Journal of Network and Computer Applications 108 (2018) 37–52

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

Attribute based encryption in cloud computing: A survey, gap analysis, and future directions

Praveen Kumar P [a,*], Syam Kumar P [b], Alphonse P.J.A. [a]

[a] Department of Computer Applications, National Institute of Technology, Tiruchirappalli, India
[b] Institute for Development and Research in Banking Technology, Hyderabad, India

ARTICLE INFO

Keywords:
Cloud computing
Attribute based encryption
Key policy
Ciphertext policy
Fine-grained access control
Revocation mechanism

ABSTRACT

Cloud computing facilitates to store and access the data remotely over the internet. However, storing the data in the untrusted cloud server leads the privacy and access control issues in the cloud. The traditional encryption schemes such as symmetric and asymmetric schemes are not suitable to provide the access control due to lack of flexibility and fine-grained access control. One of the prominent cryptographic technique to provide privacy and fine-grained access control in cloud computing is Attribute Based Encryption. In this paper, we comprehensively survey the various existing key policy and ciphertext policy attribute based encryption schemes based on access structure, and multi-authority schemes. Moreover, this review explores more on ciphertext policy attribute based encryption in different aspects such as hidden policy, proxy re-encryption, revocation mechanism, and hierarchical attribute based encryption. Further, this paper compares different ABE schemes based on the features, security, and efficiency. This paper also identifies the suitability of attribute based encryption for practical applications. Finally, this paper analyze the different ABE schemes to find out the research gap and challenges that needs to be investigated further on the Attribute Based Encryption.

1. Introduction

Cloud environment (Lumb et al., 2009; Jadeja and Modi, 2012; Fehling et al., 2014) provides the new dimension of utilizing information technology resources in the business. The cloud delivers the resources based on the on-demand and pay by use model i.e. whenever we need the additional resources based on the request, the service will be allotted and charged. The cloud delivers the variety of services such as Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) to cloud users as shown in Fig. 1. SaaS provides the application to the user such as webmail, program interface, and web browser. PaaS provides the programming languages, libraries, services, and tools, etc. IaaS provides the infrastructure, such as storage, networks, and other processing and computing resources. There are various deployment models such as private, public, community, and hybrid cloud. Private cloud is owned by a single organization, whereas the public cloud is shared by multiple consumers. Community cloud means the same kind of community consumers can join and use this service. Hybrid cloud is the combination of any two above-said deployment models of the cloud. Based on the user need and requirement, the user may choose specific services and deployment model.

Cloud provides a lot of benefits such as cost savings in investments, less maintenance, flexibility, less environment impact, scalability, access anywhere, etc. Even though the cloud provides a lot of benefits, the businesses or organizations are not moving to the cloud, especially storing big data in the cloud due to security and privacy issues (Takabi et al., 2010; Pasupuleti et al., 2016). Cloud storage (Wu et al., 2010) is mainly used to store and manage the data remotely, it allows storing the data through the internet so that users can access the data from anywhere in the world irrespective of the device and location. Cloud storage also helps to store big data for managing, processing and analyzing of data, which is quite simple without investing much because it supports the entire requirement for the same. However, the problem with cloud storage is data privacy and access control, because storing data in the cloud means it is stored in third party Cloud Service Provider (CSP) who may not trusted. Therefore, the cloud service provider may access or disclose the sensitive data and they may share the stored data to unauthorized users for business purposes. Consider the privacy and security of the users the data must be encrypted before storing in to the cloud. Even though we encrypt the data it can be accessed by all the users, so the data access should be restricted based on user's access level and rights. Henceforth, there are two main things to be considered while storing the

* Corresponding author.
E-mail addresses: tvp.praveen@gmail.com (P.K. P), psyamkumar@idrbt.ac.in (S.K. P), alphonse@nitt.edu (A. P.J.A.).

https://doi.org/10.1016/j.jnca.2018.02.009
Received 26 June 2017; Received in revised form 13 December 2017; Accepted 17 February 2018
Available online 19 February 2018

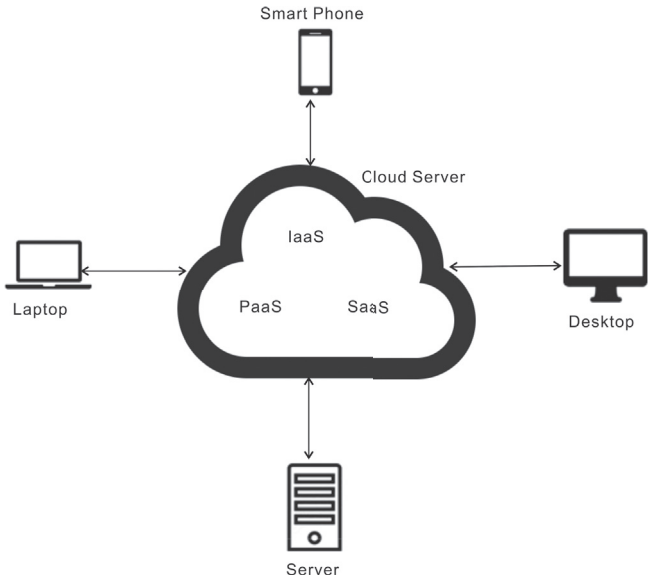

Fig. 1. Cloud computing architecture.

data in the cloud i.e. privacy of the big data and user access control (Khan, 2012).

The traditional symmetric and asymmetric key encryption cryptographic techniques are used for encryption. The symmetric key means the same key is used for both encryption and decryption. The asymmetric key means the public key is used for encryption and the private key is used for decryption i.e. different keys are used for both encryption and decryption. However, these encryption techniques provide the privacy, but not the access control. The Attribute Based Encryption (ABE) is a public key cryptographic technique (Kamara and Lauter, 2010) that provides the secure data sharing among multiple users which can achieve both privacy and access control. In ABE, data is encrypted using attributes and decrypted using the secret key of a user which is associated with an access policy. The user can only decrypt when the user credentials satisfy the access policy, and it does not only provide the fine-grained access control, but also provides revocation, collusion resistant, and scalability. The ABE is mainly classified into two types, Key Policy Attribute Based Encryption (KPABE) and Cipher Policy Attribute Based Encryption (CPABE). In KPABE, the ciphertext is based on attributes and the user's secret keys are based on access policies, and it is shown in Fig. 2. In CPABE, the ciphertext is based on access policies and user's secret keys based on attributes and it is shown in Fig. 3.

In this paper, we survey and discuss the different ABE schemes, such as KPABE, CPABE and about access structure, constant ciphertext, and multi-authority. Further CPABE is studied in-depth about hidden policy, proxy re-encryption, revocation mechanism and Hierarchical ABE. Finally, we conclude and provide the future direction of ABE research. The taxonomy of the survey is shown in the Fig. 4. The major contributions of this paper are as follows:

1) We survey the ABE schemes in cloud computing
2) We discuss the problems of different KPABE and CPABE schemes
3) We give the comparison of various ABE schemes based on their advantages, disadvantages, and functionalities.
4) We also provide the security and performance analysis of different ABE schemes
5) Finally, we provide the applications and future directions for ABE in cloud computing

The rest of the paper is organized as follows: Section 2 focus on the ABE basic concepts with an algorithm, Key policy ABE, Ciphertext policy ABE, multi-authority CPABE, hidden policy CPABE, proxy re-encryption, revocation mechanism in CPABE, and Hierarchical attribute based encryption. The comparative study of the various ABE schemes is presented in Section 3. The applications of ABE is given in Section 4. Section 5 covers the future directions of ABE research and the paper is concluded in Section 6.

2. Attribute based encryption

In this section, we discuss about the fundamental concept of ABE and its algorithm. Sahai and Waters (2005) first proposed the concept of ABE, and it is the public key cryptography of one-to-many algorithm to protect the data in the cloud. Here, the encryption of the data is based on the set of attributes. There are three actors involved namely Authority, Data Owner, Data User. Authority generates public key and send it to the data owner for encryption and it also generates the master secret key. Moreover, the authority generates the user's secret key with the master secret key according to the attributes. Data owner receives the public key to encrypt the data along with the attributes and store it into the cloud. Data

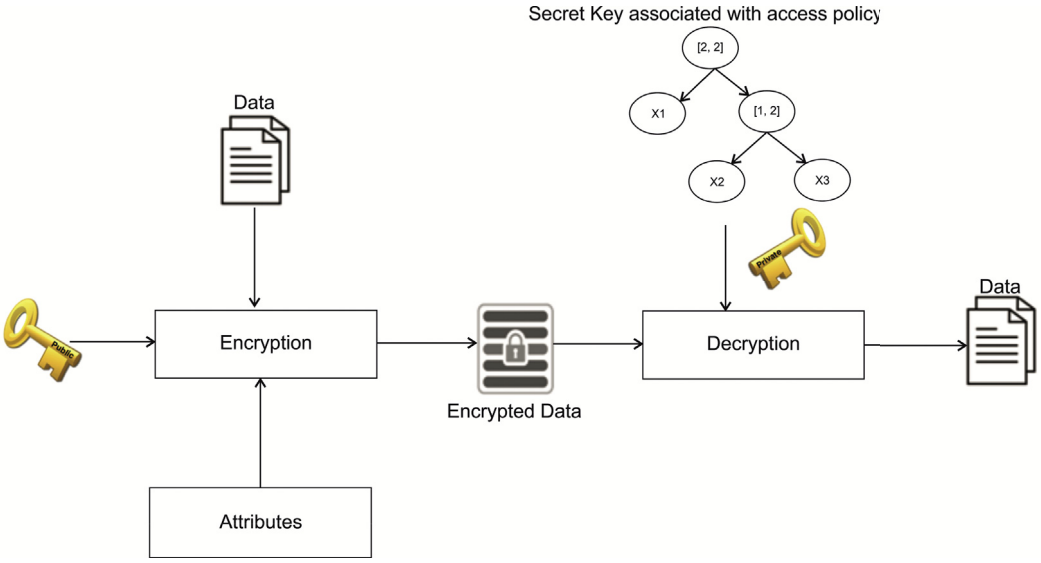

Fig. 2. Key policy attribute based encryption.

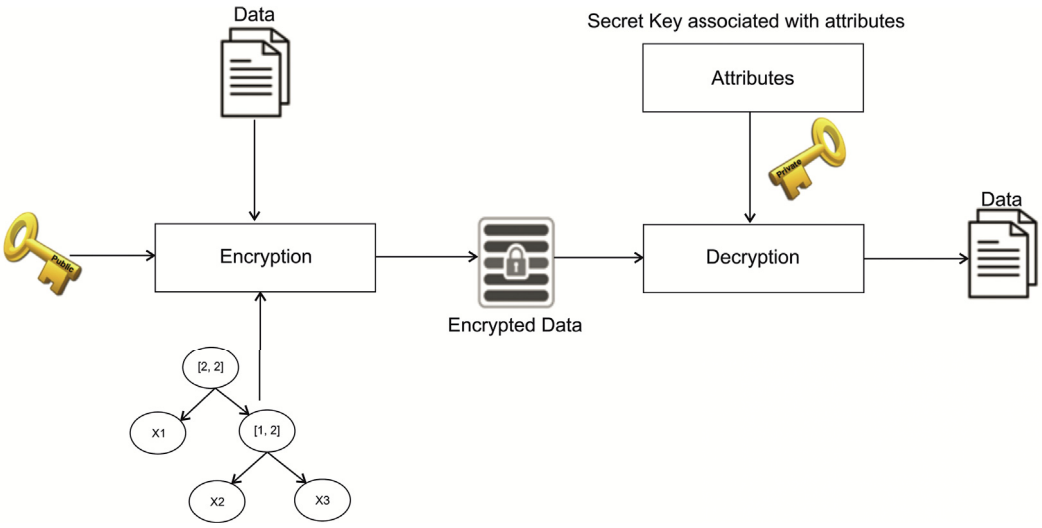

Fig. 3. Ciphertext policy attribute based encryption.

user gets the private key from authority and decrypts the data as required. The data decryption is possible only when atleast *d* component of the attributes in the encrypted data match with attributes in secret key. If any user wants to add the system, again the authority will redefine and generate the keys again. The architecture of this ABE scheme is shown in Fig. 5. This scheme contains four different algorithms. The different common notations used in all the ABE algorithms in this paper are tabulated in Table 1.

Setup: Let ℓ: G1 X G1 → G2 denote the bilinear map. The authority produces the public parameter Public Key (PK) and Master Secret Key (MSK) using non-zero random value x_1, x_2 … x_n and y from Z_p.

$$PK = ((R_1 = g^{x_1}, \ldots, R_n = g^{x_n}), Y = \ell(g,g)^y)$$
$$\text{MSK} = (x_1, x_2, \ldots..x_n, y) \quad (1)$$

KeyGeneration: Authority produces the user's secret key (S) for every user. Let q(0) = y. The user's secret key S is:

$$Sj = g^{\frac{q(j)}{x_j}}, \text{where j} \in \text{AU};\ \text{AU} \subseteq \text{UA} \quad (2)$$

Encryption: This algorithm produces the ciphertext. The message (M) is encrypted using PK and set of attributes (ω) used as an identity by data owner. The encrypted text E is defined as:

$$E = \left(\omega, E' = MY^c = \ell(g,g)^{yc}, \left\{E_j = R_j^c\right\} j \in \omega\right) \quad (3)$$

Decryption: Data user decrypts the message from ciphertext using secret key S. S generated with the identity (AU) and ciphertext is generated with the identity (ω). The decryption is possible only if $|\text{AU} \cap \omega| \geq$d. Compute the message M as:

$$M = E'/Y^c \quad (4)$$

The ABE is mainly classified in to KPABE and CPABE. We review the KPABE and CPABE schemes in the forthcoming sections.

2.1. Key policy attribute based encryption

Here, we discuss about the different KPABE schemes. Goyal et al. (2006) introduced the KPABE, which is a cryptosystem for fine-grained sharing of encrypted data. In this scheme, the ciphertext is formed with the set of attributes and user key is embedded with policy i.e. access structure. The user can decrypt the message when the user attributes satisfy the access structure. A tree based access structure was used here, which is an access tree where the leaf nodes consist of attributes and non-leaf nodes consists of threshold gates in the form of [*m, n*] where *m* is threshold value and *n* represents the number of attributes followed. If the threshold is [*1, n*], then it represents the OR gate and [*n, n*] represents the AND gate. If the root node is satisfied, then the user can decrypt the data otherwise it is not possible to decrypt the data. Let take X1, X2, X3 are attributes and the threshold gate tree structure for the access policy X1 AND (X2 OR X3) which is shown in Fig. 6. Here, [*1, 2*] represents any one of the attributes among X2 or X3 attributes i.e. OR gate and [*2, 2*] represents AND gate. In this scheme, the same four algorithms explained in the ABE are used, but there is a change in keygeneration and decryption because of access structure involved in the both algorithms. The algorithms described as follows:

Setup: The authority produces the public parameter Public Key (PK) and Master Secret Key (MSK) using non-zero random values t_1, t_2 … t_n and y from Z_p.

$$PK = ((R_1 = g^{t_1}, \ldots, R_n = g^{t_n}), Y = \ell(g,g)^y)$$
$$\text{MSK} = (t_1, t_2, \ldots..t_n, y) \quad (5)$$

KeyGeneration: Authority generates the secret keys (D) for every user using MSK, and Access structure. User can decrypt the data provided τ(AU) = 1. Let $q_x(0) = q_{\text{parents(x)}}$ (index(x)) and $q_{r0}(0) = y$, The private key D defined as:

$$D_x = g^{\frac{qx(0)}{tj}}, \text{where } j \in AU; \quad (6)$$

Encryption: The message (M) is encrypted using PK and set of descriptive attributes (ω). The encrypted text E is defined as:

$$E = \left(\omega, E' = MY^c = \ell(g,g)^{yc}, \left\{E_j = R_j^c\right\}\right), j \in \omega \quad (7)$$

Decryption: Data user decrypts the message from ciphertext using Secret key. Compute $M = \ell(E_j, D_x) = (g,g)^{cq_x(0)}$ if j ∈ AU. The ciphertext can be decrypted as:

$$M = E'/Y^c \quad (8)$$

Monotonic access structure is used in this scheme which means access policy does not have the negative attribute. Ostrovsky et al. (2007) introduced the non-monotonic access structure which includes the both positive and negative attributes. It supports AND, OR and NOT between attributes, namely X1, X2, X3 are attributes and the non-monotonic tree access structure is shown in Fig. 7 for the access policy X1 AND X2 NOT X3.

However, non-monotonic access policy is more complex access structure compared with monotonic-access structure, and also it suited

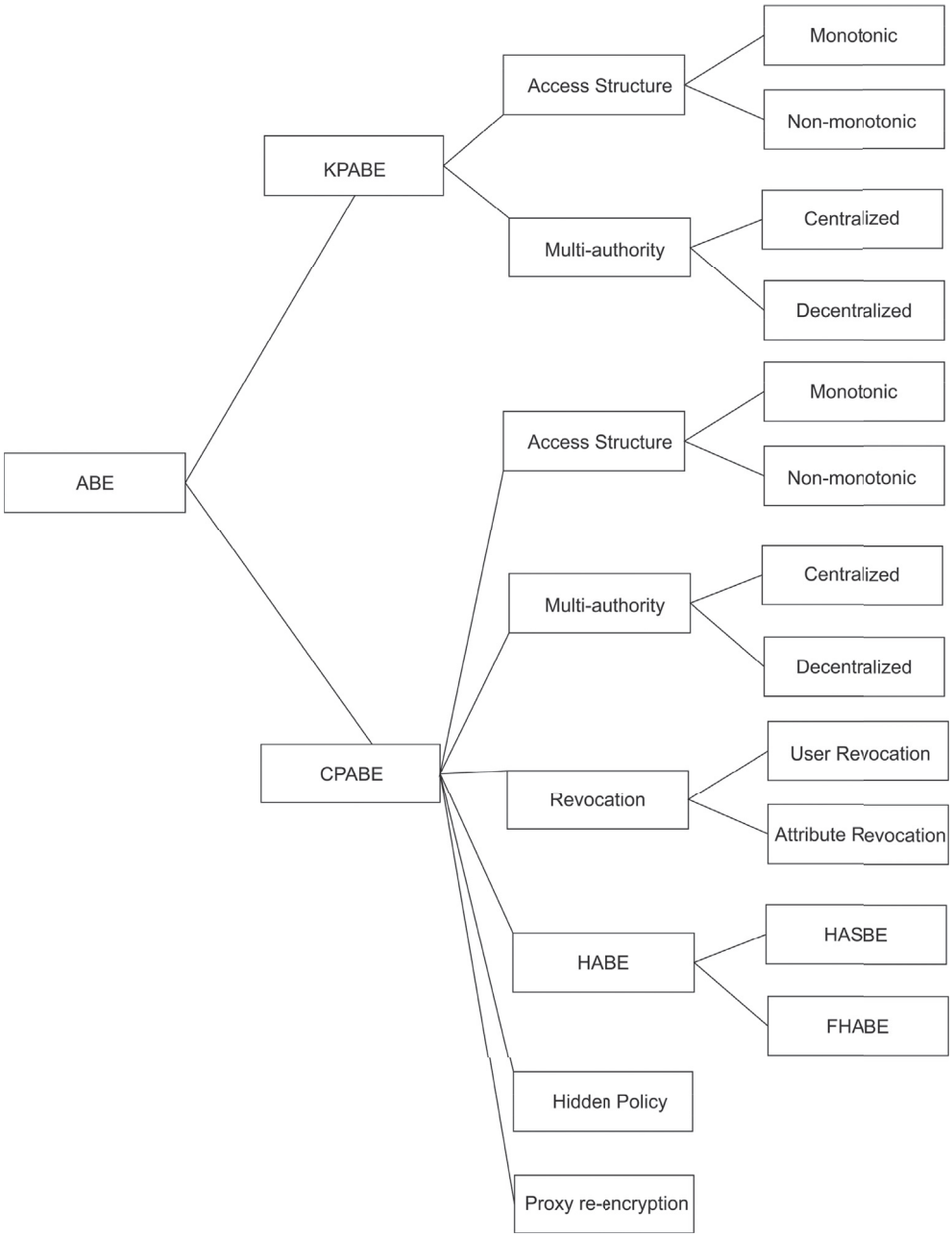

Fig. 4. Taxonomy of ABE survey.

only for fixed number of attributes and bounded number of users. Moreover, the size of the ciphertext, secret key and computational overheads of encryption and encryption process is high. Lewko et al. (2010) improved the non-monotonic access structure policy to achieve the user revocation with the small size of private keys. It supports unbounded users. In all the previous work (Goyal et al., 2006; Ostrovsky et al., 2007; Lewko et al., 2010), researchers only focused on access structures represented a Boolean formula, but they never focused on the growth of the ciphertext size. The size of ciphertext text is increased linearly depends on the number of attributes available in the policy. Attrapadung et al. (2011) proposed the non-monotonic access structure with constant ciphertext size. Constant size ciphertext means the size of ciphertext is not dependent on the number of attributes rather it depends on the security parameter. This scheme reduces the number of pairing during decryption also. Usually, one pairing operation required per attribute. Wang and Luo (2012) proposed the constant size ciphertext for monotonic access structures. This work based on Delerablee (2007) identity based broadcast encryption scheme. Both the works (Attrapadung et al., 2011; Wang and Luo, 2012) are selectively secure but not fully secure. Later, Lai et al. (2014) proposed the work of constant size ciphertext for monotonic access structures with fully secure and fast decryption. This scheme is also reduced the number of pairing during the decryption phase.

The schemes (Attrapadung et al., 2011; Wang and Luo, 2012; Lai et al., 2014) proposed the constant size ciphertext but the size of secret keys was quadratic size with respect to the number of attributes. Moreover, in all the previous works, the single authority is used. In single authority scheme, the authority can decrypt all the ciphertext because authority is going to issue all the secret keys. Chase (2007) solved the single authority problem by introducing the multi-authority ABE scheme. Here, the data owner has to specify the threshold value (d) and set of attributes for each authority. The user can decrypt the message, if and only if the user has at least d number of given attributes from each authority. In this scheme, every user has some Global Identifier (GID), it

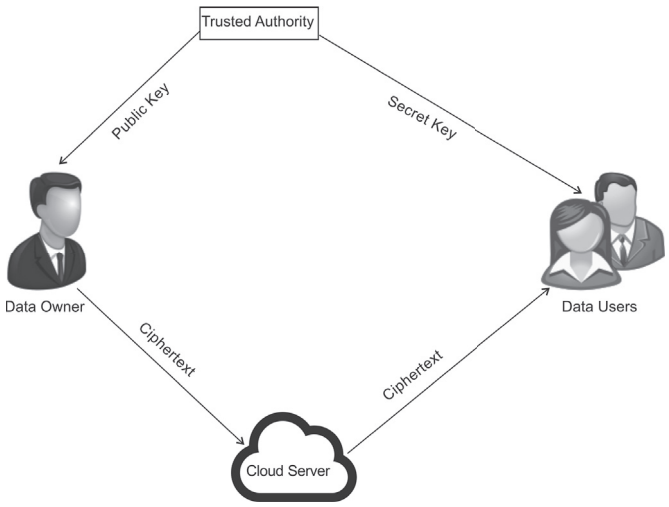

Fig. 5. Architecture of attribute based encryption.

may be name or SSN or any identifying string. There is a Central Authority (CA), where each user sends the GID to CA and receives a corresponding secret key. The authority will not get any information about user's attributes, and this architecture is shown in Fig. 8. However, the problem here is the CA has the power to decrypt all ciphertext, moreover, the use of consistent GID allowed the authorities to combine their information to build all users' attributes, so the privacy of data is compromised here. In their subsequent work, Chase et al. (Chase and Chow, 2009) proposed another multi-authority scheme which does not have the central authority and protecting the user information from the

Table 1
Different notations used in various ABE algorithms.

Notation	Meaning
p	prime order
G1, G2	bilinear group of prime order p
g, g1	generator of the group
d	threshold value
UA	universe of attributes
n	number of attributes in the UA
q, q_x	random polynomial of degree d-1
AU	set of user attributes
c, r	non-zero random values from Z_p
r0	root node of the access tree
x	node of the access tree
τ	access tree

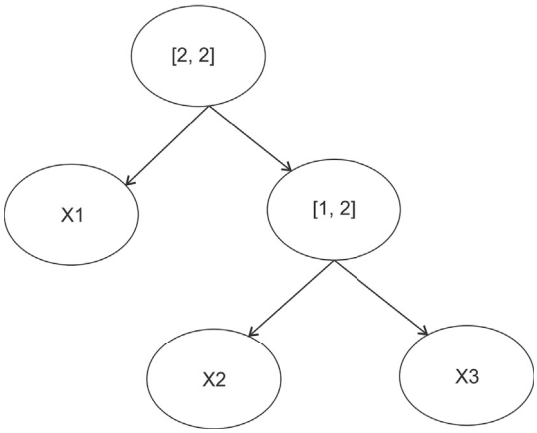

Fig. 6. Threshold gate access structure.

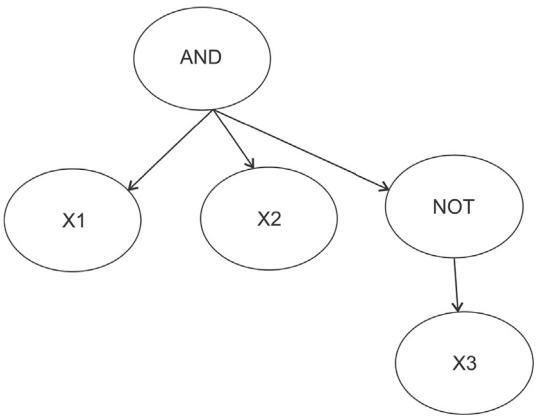

Fig. 7. Non-monotonic access structure.

pooling of information on particular users. In this scheme, key issuing protocol used which allows the users to communicate with authority via pseudonyms instead of having to provide their GIDs and it also prevents the authority from pooling their information. However, multiple authorities must collaborate to setup the system. The system consists of number of authorities (N), and if more than N-2 authorities are corrupted, then the system is not secure.

Next, the decentralized key policy attribute based encryption introduced by Han et al. (2012) and the architecture is shown in Fig. 9. Here multiple authorities need not be online always. This scheme eliminates the heavy communication cost. Every authority can join or leave the system at any time freely. Authorities can issue the secret key without communicating among themselves and GID. The authors used the privacy preserving key extraction protocol to achieve the security in standard complexity assumption. The problem with this scheme is that it cannot prevent user collusion, i.e. two users can pool their decryption keys to generate decryption keys. Rahulamathavan et al. (2016) improved the Han et al. (2012) work and proposed a user collusion avoidance scheme for privacy preserving decentralized key-policy ABE. Here the main work was tieing the secret known for authority and secret known for the user in a non-linear fashion. This scheme relies on standard complexity assumption.

Hence, KPABE reduces the computational overhead in the cloud server, but the problem with the KPABE is that the data owner doesn't have the rights or control to decide who can access the data which restrict the possibility and usability for the system in practical applications.

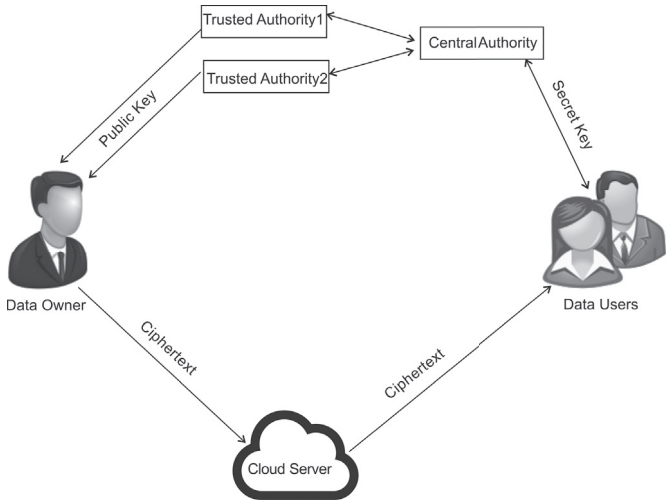

Fig. 8. Architecture of centralized multi-authority scheme.

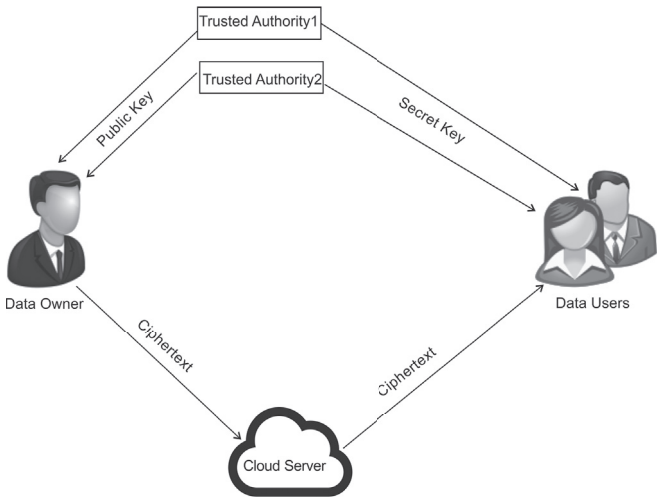

Fig. 9. Architecture of decentralized multi-authority scheme.

2.2. Ciphertext policy attribute based encryption

Here, we discuss the different CPABE schemes. Bethencourt et al. (2007) first introduced the ciphertext policy attribute based encryption. The ciphertext is associated with access structure i.e. policy and user's secret keys are associated with attributes in CPABE. A monotonic threshold gate access structure was used. The security was proved using generic group model. In this scheme, five algorithms used;

Setup: The authority generates the public parameter Public Key (PK) and Master Secret Key (MSK) using non-zero random value α, β from Z_p.

$$PK = \left(G_1, g, h = g^{\beta}, f = g^{1/\beta}, \ell(g,\ g)^{\alpha}\right), \quad \text{MSK} = (\beta, g^{\alpha}) \tag{9}$$

KeyGeneration: Authority generates the secret keys (S) for every user using MSK and AU. Generate non-zero random value r_a for each attribute a∈ AU. The private key S is defined as:

$$S = \left(g^{\frac{\alpha+r}{\beta}}, \forall a \in AU; D_a = g^{r}.H(a)^{r}_{a}, D'_a = g^{r}_{a}\right) \tag{10}$$

Encryption: The message (M) is encrypted using PK and access structure (τ). Let $q_x(0) = q_{parents(x)}(index(x))$ and $q_{r0}(0) = s$, where $s \in Z_p$. Let LN represents the set of leaf nodes. The encrypted text E is defined as:

$$\tilde{E} = M\ell\ (g,g)^{\alpha s},\ E_1 = h^{s}, \forall i \in LN,\ E_y = g^{q_i(0)},\ E'_y = H(att(i))^{q_i(0)} \tag{11}$$

Decryption: Data user decrypts the message from ciphertext using secret key. Let v=w = att(x) and compute M if w∈ AU as;

$$M = \tilde{E} \Big/ \frac{\ell(\mathrm{D_w}, \mathrm{E_v})}{\ell(\mathrm{D_{w'}}, \mathrm{E_{v'}})} = \tilde{E} \Big/ \frac{\ell(\mathrm{g^r}.\ \mathrm{H(w)}^{\mathrm{r_a}}, \mathrm{g}^{\mathrm{q_v}(0)})}{\ell(\mathrm{g^r}, \mathrm{H(w)}^{\mathrm{q_v}(0)})} = \tilde{E} \ / \ \ell(g,g)^{rq_v(0)} \tag{12}$$

Delegation: It takes the input as user private key (S) and regenerate the new key whenever updates required which is equivalent to the key generation of authority. Let $\overline{A}$ be another set of attributes, where $\overline{A} \subseteq$AU. Choose non-zero random value $r_w' \in Z_p$ for each attribute w∈ AU. The newly generated private key $\tilde{S}$is:

$$\tilde{S} = \left(Sf^{\tilde{r}}, \forall w \in \overline{A}: \tilde{S}_w = S_w.g^{\tilde{r}}.H(w)^{\tilde{r}_w}, \tilde{S}_w = S'_w.g^{\tilde{r}_w}\right) \tag{13}$$

Cheung and Newport (2007) improved the security proof of Bethencourt et al. (2007) work and it proved that the chosen ciphertext secure under the Decisional Bilinear Diffie-Hellman (DBDH) assumption first time. It used AND gate positive and negative attributes as access structure. They used don't care condition to find the attribute which is not in the AND gate. This scheme was not much expressive because it used only AND gate and also, the ciphertext and secret key size increased linearly with the number of attributes. Goyal et al. (2008) presented the improved version of CPABE which supports the bounded size access tree with threshold gate and security proved the under the standard model, decisional Bilinear Diffie-Hellman assumption. The problem with this scheme was the depth of the tree should specify in the setup phase itself. So the user may be restricted to use only an access tree which has the depth of the tree less than depth which is specified in the setup phase. Liang et al. (2009a) improved the scheme (Goyal et al., 2008) by providing the faster encryption, decryption and shortened the ciphertext size. This scheme was proved under DBDH assumption. Ibraimi et al. (2009a) proposed the efficient CPABE policy that can express any access policy with AND & OR Boolean operator along with threshold. Here, the access structure is an n-ary tree which consists of AND, OR and the threshold value are as intermediate nodes and attributes are in leaf nodes. It proved the security under DBDH assumption. The main practical problems in the above CPABE schemes were attribute revocation. In the sub-sequent works, Ibraimi et al. (2009b) resolved the above-said problem and proposed the CPABE scheme with attribute revocation.

The above schemes (Bethencourt et al., 2007; Cheung and Newport, 2007; Goyal et al., 2008; Liang et al., 2009a; Ibraimi et al., 2009a; Ibraimi et al., 2009b) produced the ciphertext size which is either complex or liner with respect to the number of attributes. Emura et al. (2010) proposed the constant length ciphertext scheme in which they have shown that the number of pairing computation is also constant. AND gate with multi attributes access structure was used in this scheme. Later, Waters (2011) proposed a new scheme that expresses access control by Linear Secret Sharing Scheme (LSSS) matrix over attributes. However, the size of ciphertext and encryption and decryption overheads increased linearly. Recently, Li et al. (2017) improved the CPABE by introducing new access structure Ordered Binary Decision Diagram (OBDD). It is a non-monotonic access structure, and it supports AND, OR and NOT between attributes. They had Boolean variables X1, X2, …, Xn making up the input to a function. Here each Boolean variable represent the attributes. Each Boolean variable takes two children, left child and right child, each of the children may have their child and so on. All 0 value subtree edges are connected with dotted lines and 1 value subtree edges are connected with solid lines. At the leaves, they had either 0 or 1, which is the output of the function on the inputs that constitute the path from the root to the leaf. Start traversal from root node to leaf node based on the attribute value, at the end of the traversal the leaf node value determine the access. Let take X1, X2 are attributes and we have two edges for X1 = 0 and X1 = 1. Each of the two subtrees is now testing another variable, each with another two subtrees, and so on. The binary decision diagram representation of the access policy (X1 V X2) shown in Fig. 10. It can reduce by two ways. The first way is to test whether any redundant Boolean variables are available, if so omit those Boolean variables. The second way is finding out identical subtrees, if so, allow them to share. After reducing the Boolean variable order the tree again. This access structure is called ordered binary decision algorithm. The reduced version of the (X1 V X2) access tree structure shown in Fig. 11.

All the above CPABE schemes (Bethencourt et al., 2007; Cheung and Newport, 2007; Goyal et al., 2008; Liang et al., 2009a; Ibraimi et al., 2009a; Ibraimi et al., 2009b; Emura et al., 2010; Waters, 2011; Li et al., 2017) were used only single authority. In the single authority scheme, the authority produces the all private key for users which is required for decrypting the data. Here escrow problem arrives i.e. the authority can decrypt all the ciphertext because authority is going to issue all the secret keys. In some situation different kind of users requires different attributes set. This scenario was not possible to handle by single authority. For example, IBM course can be conducted directly by IBM organization or any affiliated or authorized educational institutions. All have a different kind of users, and a single authority is not providing the effective solution for this use case. The solution is multi-authority CPABE. Lewko and Waters (2011) first proposed a new multi-authority CPABE scheme, where no

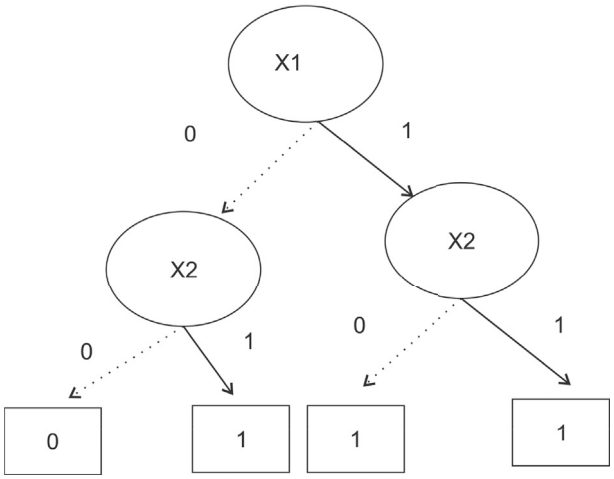

Fig. 10. Representation of access policy X1V X2 using binary decision diagram.

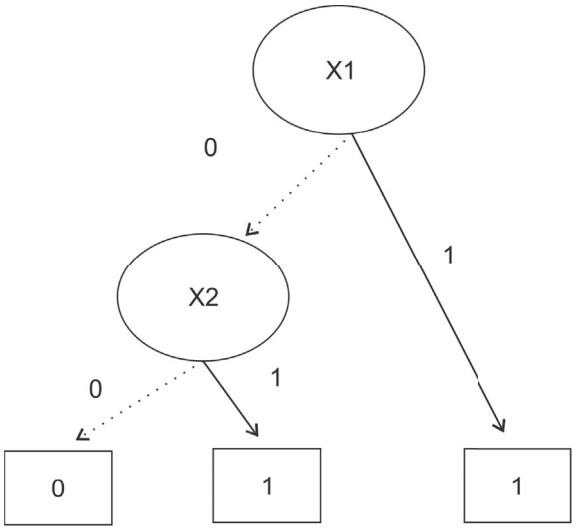

Fig. 11. Reduced version of the access policy X1 V X2 using binary decision diagram.

central authority is available and the authorities can able to add or quit the system without any constraints. There are five algorithms used:

Global Setup(λ) - > GP: Let G be the bilinear group of composite prime order N = p1p2p3. Let H is the hash function, and this maps global identities (GID) to elements; H: {0, 1}* - > G. The global setup algorithm outputs global parameter GP which consists of N and generator g1.

Authority Setup(GP) - > SK, PK: Authority setup algorithm produces the public parameter, public key (PK) and master secret key (MSK) with GP as input and non-zero random value α_a, y_a in Z_N. This algorithm runs for each authority in the system. Let A_a represents authority's attributes.

$$PK = (\ell(g_1, g_1)^{\alpha_a}, g_1^{y_a} \forall a \in A_a), MSK = (\alpha_a, y_a \forall a \in A_a) \tag{14}$$

Encrypt(M, (AM, ρ),GP, {PK}) - > E : The message (M) is encrypted using relevant PK, an access matrix AM with ρ mapping its rows to attributes, and GP as input. Let λ_x denotes AM_x.V, where x is the row, V ∈ Z_N and Let ω_x denotes $AM_x.\omega$, where ω ∈ Z_N. The encrypted text E is defined as:

$$\begin{aligned} E_0 &= M\ell(g1, g1)^c, \\ E_{1,x} &= \ell(g1, g1)^{\lambda_x}\ell(g1, g1)^{\alpha_{\rho(x)} r_x}, \\ E_{2,x} &= g1^{r_x}, \\ E_{3,x} &= g1^{y_{\rho(x)} r_x} g1^{\omega_x} \forall x \end{aligned} \tag{15}$$

KeyGeneration(GID,GP,a,SK) - > $Da_{,GID}$: KeyGeneration algorithm produces the user's secret key $D_{i,GID}$ using identity GID, GP, attributes A_a which is belonging to concern authority, and the master secret key MSK for the authority as input. Each authority generates the key for their concern users only.

$$D_{i,GID} = g_1^{\alpha_a} H(GID)^{y_a} \ \forall a \in A_a \tag{16}$$

Decrypt(E,GP,{$D_{i,GID}$}) - > M: Data user decrypts the message from ciphertext using secret key $D_{i,GID}$. It also takes global parameters as input. The decryption is possible only when, the collection of attributes user attributes that satisfies the access matrix corresponding to the ciphertext. The message can obtain by

$$M = E_0/\ell(g1, g1)^c \tag{17}$$

The drawback of this scheme is low efficiency and user's attributes might be found by tracking his/her global identifier. Liu et al. (2011) proposed a new multi-authority CPABE, which consists of multiple central authorities. CA issues the identity related keys to user and authorities issue the attribute related keys to the user. Authorities are working independently with each other and it used the monotonic access structure. The security was proved under the standard assumption, but this scheme was also less efficient. Li et al. (2011a) improved the multi-authority CPABE scheme with user accountability, which reduces both the trust assumption on the authorities and the users. The security of standard model was proved under DBDH assumption, the Decisional Linear (DLIN) assumption and the q-Decisional Diffie-Hellman (q-DDH) in-version assumption. In multi-authority schemes, the entire attribute set is divided into multiple disjoint sets and each disjoint set is assigned to each authority, but still the problem of single authority exists. To overcome the problem, Li et al. (2016a) proposed the threshold multi-authority CPABE access control scheme. In this scheme, the multiple authorities join together and manage the whole attribute set, but no one authority has the full control of any attribute, so the master key cannot be obtained by any one authority alone.

2.2.1. *Hidden policy*

The hidden policy means access structure is hidden. While sending the ciphertext to the cloud, the policy is sent along with the ciphertext; therefore, anyone who accesses the ciphertext can know the access policy. Thus, the user may learn about the policy, so it leads to the weak policy privacy. Nishide et al. (2008) proposed the new hidden policy to overcome the above-said problem. Let *n* be the number of attributes in the system and the set of possible value of attributes are denoted as $AV = \{AV_1, AV_2, \ldots, AV_n\}$. The possible values for attributes are binary numbers 0, 1 and wild card value *. Each attribute may take more than one value. Let AL be a user attribute value list, and it is denoted as $AL = \{L_1, L_2, \ldots, L_n\}$, $L_i \in AV$. Let *C* be the access structure for ciphertext, and it is denoted as $C = \{C_1, C_2, \ldots, C_n\}$, $C_i \subseteq AV$. The user can decrypt the data provided if it satisfies the condition $L_i = C_i$ or $Ci = *$, for i = 1 to n. Let's take the example of 5 attributes and possible values of attributes are AV = {0, 1, *}. The access structure is specified as C = {1, *, 1, 0, *}, that means the value of the A1 and A3 is 1, and value of A4 is 0, and A2 and A5 are don't care values. The user secret key associated with the value {1, 1, 1, 0, 0} can decrypt but {1, 1, 0, 0, 1} cannot decrypt the ciphertext. In case of more than one value for single attribute, they used Boolean operators AND & OR. In this manner, they achieved the policy hiding. They used, AND gate on multi-valued attributes with wildcards access structures, and used inner product predicate encryption technique to achieve this hidden policy. This model was proved as selectively secure. The basic four algorithms of CPABE (Bethencourt et al., 2007) were used here.

Lai et al. (2011) improved the Nishide et al. (2008) scheme and proposed the fully secure hidden policy with the same access structure used in (Nishide et al., 2008). The problem with this scheme is that the size of the ciphertext grows based on the number of attributes, and it

incurs the higher computation cost. Li et al. (2009) proposed the selectively secure hidden policy with shorter public parameters and ciphertext length. Security was proved under DBDH and DLIN assumptions and this scheme used AND gate on multi-valued attributes with wildcards access structures. In the subsequent work of Li et al. (2012), they proposed the fully secured scheme and security was proved under DBDH assumptions. Phuong et al. (2016) proposed the new scheme which overcomes the problem of the size of the ciphertext in hidden policy (Lai et al., 2011). In this scheme, the size of the ciphertext is constant and along with the hidden policy, it used the AND gate with positive, negative and wildcard access structure is also used. This scheme is selectively secured. Jin et al. (2016) proposed the fully secure hidden policy with short ciphertext size and AND gate with positive, negative and wildcard were used in the access structure.

2.2.2. Attribute based proxy re-encryption

Attribute based proxy re-encryption is that the data owner delegates his capability to proxy for re-encrypting the data according to the new access policy to maintain effective access control while an owner is offline. Let's discuss an application scenario, in university, the faculty handling the subject cryptology requires to maintain the students' subject related information securely, so he encrypts with access policy FACULTY AND CRYPTOLOGY and stores it in the cloud. He may decrypt the ciphertext whenever required. In case, if he is going on vacation or taking days off, if the stored subject information need to be access during his absence then he must share the information with another faculty member. New faculty can read the information only if his attributes matched with access policy associated with the ciphertext. So we have to re-encrypt the ciphertext according to the new access policy where the new faculty is able to decrypt. This task is done by the faculty designated proxy that translates the ciphertext into new ciphertext with new access policy according to the new faculty, and new faculty can decrypt the re-encrypted ciphertext. This is the principle of Attribute based proxy re-encryption (ABPRE), which combine traditional proxy re-encryption with the attribute based component. Liang et al. (2009b) proposed the first CP-ABPRE scheme, in which the data owner has delegated his capacities to proxy in the data access control when he is offline. Proxy re-encrypts a ciphertext according to the new policy. The master key was generated without using random oracle, and it was the selectively secure model under Augment Decisional Bilinear Diffie-Hellman assumption (ADBDH). It consists of six algorithms, among six, four algorithms setup, keygeneration, encryption, and decryption are same as in CPABE and two new algorithms are as follows:

RKGEN(S, AS)->(rk): This algorithm generates the user re-key *rk* using secret key and an access structure.

REENC(rk, CT) -> (CT′): This algorithm re-encrypts the ciphertext CT and produces the new ciphertext CT′ using the input *rk* and a ciphertext CT. This algorithm first checks if the index set in *rk* satisfies the access structure of CT. If it matches, then it produces the output CT'; otherwise it rejects it.

Luo et al. (2010) extended the previous scheme and introduced the re-encryption control i.e. the encryptor can decide whether the ciphertext can be re-encrypted or not. This scheme used the AND-gate multivalue attribute, negative and wildcard access structure. It was selectively secure under the decisional bilinear Diffie-Hellman assumption. Both the schemes (Liang et al., 2009b; Luo et al., 2010) suffered from the computational cost with the number of pairing operations that is required. Seo and Kim (2012) proposed a scheme to reduce the computation overhead by minimizing the number of pairing operation with exponent operation. Here AND-gate positive, negative access structure was used. It was selectively secure model under augment decisional bilinear Diffie-Hellman assumption. Liang et al. (2013) proposed the variation in the existing schemes and used the random oracle model. It was suitable for any monotonic access structures. The security model is proved using Decisional q-parallel Bilinear Diffie-Hellman Exponent (D q-parallel BDHE) assumption.

Li (2013) proposed the variations in the proxy re-encryption with LSSS matrix access structures. The security was proved under decisional parallel bilinear Diffie-Hellman exponent assumption. In the subsequent work, Liang et al. (2015) introduced the more efficient method using dual system encryption and security was proved as selectively secure model under augment decisional bilinear Diffie-Hellman assumption. It supports any monotonic access structures. Xu et al. (2016) improved the efficiency using weighted access tree structures which consist of AND, OR and threshold gate as inner nodes and it was reduced the computational cost. The security is proved using DBDH security assumptions.

2.2.3. Revocation mechanism

In CPABE schemes, there may be a chance of dynamically changing the user's attributes because of expiring of attributes, revoke of attributes or need of adding new attributes. This type of revocation mechanism is called attribute revocation. Similarly, there is the chance in the change of user because of un-trusted or removed or added users. This revocation mechanism is called user revocation. There are two different ways of providing the solution to the revocation mechanism, direct and indirect method (Attrapadung and Imai, 2009).

2.2.3.1. Indirect method. Indirect method means revocation mechanism by authority, which updates the key periodically or dynamically during the occurrence of the revocation event. Yu et al. (2010) proposed the attribute based revocation scheme which integrated the proxy re-encryption method to achieve the revocation. During the revocation, the authority re-generates the master key and generates and sends the re-encryption key to the proxy. It also updates the user's secret key and communicates to all the users. Proxy re-encrypts the ciphertext with the re-encryption key and stores it into the cloud. The authority can revoke any user's attribute at any time. All updated keys were tagged with the current version of master key revision. The only constraint is that the proxy server should be online at all times. In this scheme, there are seven algorithms used, among 7, four algorithms setup, keygeneration, encryption, and decryption are same as mentioned in CPABE (Bethencourt et al., 2007). Other three algorithms are given here:

ReKeyGen(γ,MK): This algorithm generates the new public parameter public key (PK′) and master secret key (MSK′) as an output. The generation of new public key id is delegated to the proxy servers. The new master secret key is produced with the input of present master secret key and set of attributes (γ) that needs to be updated. Whenever there is an update, the version (ver) is increased by 1. The re-key for each attribute is computed as $rk_j = t'_j/t_j$, where t_j' is the non-zero random value chosen from Z_p, $j \in \gamma$. The output of this algorithm rk is (ver, $\{rk_j\}$ $1 \le j \le 2n$).

ReEnc(CT, rk, β): This algorithm produces the new ciphertext CT′ with existing ciphertext CT, set of proxy re-keying *rk*, and the set of attributes β which includes all the attributes in CT's access structure with proxy re-key not being 1 in *rk*. For each $j \in \beta$, $E'_j = E^{rk_j}$ *if* $1 \le j \le n$, or $E'_{j-n} = (E_{j-n})^{rkj}$ if $n < j \le 2n$. For each $j \in U$, $E'_j = E_j$ if $j \notin \beta$ and $j + n \notin \beta$, or $j \notin I$. The new ciphertext is computed by

$$E' = \left(ver + 1, AS, \widetilde{E}, \widehat{E}, \{E'_a\}_{a \in U} \right) \tag{18}$$

ReKey($\overline{D}$, rk, θ) This algorithm generates the new secret key with existing user's secret key $\overline{D}$, set of proxy re-keying *rk* and set of attributes θ which includes all the attributes in SK with proxy re-key not being 1 in *rk*. For each $j \in \theta$, $D'_j = D'_j = D_j^{rk_j^{-1}}$ if $1 \le j \le n$, or $D'_{j-n} = D_{i-n}^{rk_j^{-1}}$ if $n < i \le 2n$. For each $j \in U$, $D'_j = D_j$ if $j \notin \theta$ and $j + n \notin \theta$. While updating the version (ver) is increased by 1. The updated user secret key is computer by

$$\overline{D}' = \{D'_j, F_j\} j \in U \quad (19)$$

This scheme is selectively secured under DBDH assumption. However, this scheme failed to provide the fine-grained user access control in the data outsourcing environment. Hur and Noh (2011) provided the solution for fine grained user access control and this scheme achieved both attribute revocation and user revocation using selective group key distribution method. In his further work (Hur, 2013), he improved the security and efficiency in the fine-grained access control. Fan et al. (2014) proposed the new scheme which supports the dynamic user membership and arbitrary-state attributes. Users are able to join or quit the system without any constraints; moreover, the user can change the values of the attributes. However, the above schemes (Hur and Noh, 2011; Fan et al., 2014) were designed for single authority attribute based encryption with revocation support. Yang et al. (2013) constructed the first centralized multi-authority CPABE with attribute revocation scheme. In their subsequent work (Yang and Jia, 2014), they improved the security and efficiency of previous work. Chen et al. (Chen and Ma, 2014) proposed the user revocation using proxy re-encryption technique in decentralized multi-authority ABE schemes which do not require central authority and this scheme is based on LSSS access structure. This scheme was also addressed in the stateless receiver problem i.e. the user may not notice many key updates, so they stored all versions of update keys in the cloud store. It required the less communication cost and computation cost. The problem with this scheme was collision attack i.e. unrevoked user can share the updated key with revoked user, so the revoked user can also update their secret key and decrypt the message. Li et al. (2016b) proposed the new multi-authority CPABE scheme that supports the attribute revocation and decryption outsourcing. The security was proved under composite order bilinear groups. In this scheme, the authority produces the updated key when revocation occurs and re-encryption process is outsourced to the cloud server and it also overcomes the collusion attack problem. Wei et al. (2016) proposed the secure and efficient ABE access control for multi-authority cloud storage with revocation and it was a dynamic user revocation scheme. The overhead of revocation is linear in the logarithm of the number of revoked users, which is more efficient than the previous revocation mechanism (Yang et al., 2013; Yang and Jia, 2014). The public parameter is unchanged during revocation and it was used for the time rekeying method to handle revocation. Chow (2016) introduced a framework for multi-authority ABE with outsourcing and revocation using the same principles of (Yu et al., 2010).

The main advantage of the indirect method is the data owner need not know about the revocation details. The disadvantage of the indirect method is high communication cost and computation cost and the authority must send it to all non-revoked users for their secret key update.

2.2.3.2. Direct method. Direct method means the data owner specified the revocation mechanism during the encryption process. Liang et al. (2010) proposed the CPABE with user revocation in which they used the linear secret sharing and binary tree techniques, and it was proved the security under standard model. Here each user assigned with the unique identifier, so it is very easy to revoke the user with the unique identifier. Every timestamp (*t*), the new key was generated with revocation list which is maintained by the data owner. Addition to the general four algorithm setup, keygeneration, encryption, decryption in CPABE (Bethencourt et al., 2007), here key update is used. The algorithm is as follows:

KUpd(*rl, t, MK*) This algorithm takes a revocation list *rl*, a time stamp *t* and the master key *MK* as input. It outputs the update information *UI*. The algorithm chooses *t*, $r_{t,y} \in Zp$, and outputs the updated information $UI = \{E_y, e_y\}_{y \in KUN(r,t)} = \{B^{a_y t} H(t)^{r_{t,y}}, g^{r_{t,y}}\}_{y \in KUN(rl;t)}$ where $r_{t,y} \in Zp$ are random numbers.

Table 2
Comparison of various CPABE revocation schemes.

Scheme	Multi-authority	Attribute Revocation	User Revocation	Method
Liang et al., 2010	No	No	Yes	Time rekeying
Yu et al., 2010	No	Yes	No	Proxy Re-encryption
Hur and Noh, 2011	No	Yes	Yes	Selective group key distribution
Wu and Zhang, 2012	No	Yes	No	LSSS Matrix
Fan et al., 2014	No	Yes	Yes	Update Key
Yang et al., 2013	Yes	Yes	No	Update Key
Yang and Jia, 2014	Yes	Yes	No	Update Key
Chen and Ma, 2014	Yes	No	Yes	Proxy Re-encryption
Chow, 2016	Yes	Yes	No	Proxy Re-encryption
Li et al., 2016b	Yes	Yes	No	Proxy Re-encryption
Wei et al., 2016	Yes	No	Yes	Time rekeying

Wu and Zhang (2012) proposed the ABE scheme with attribute revocation method using direct method. The advantage of the direct method is that the data owner does not have to update key. The disadvantage of this method is that the data owner must have the revocation list, so the data owner must manage it and it becomes a complex task for the for data owner.

Revocation mechanism is the major issue in P2P cloud storage also. He et al. (2014) achieved the fine-grained access control with user revocation scheme on P2P cloud storage. In this scheme, the owner delegate the file re-encryption task to cloud servers and user's secret key is updated by reputable system peers without disclosing data contents and valid user secret keys. The comparison of various revocation schemes is tabulated in Table 2.

2.2.4. Hierarchical attribute based encryption

CPABE is a promising cryptography technique to provide fine-grained access control and data encryption. But CPABE schemes are not optimal in large enterprises because it does not support the full delegation mechanism. The full delegation mechanism is required with in enterprise itself i.e. delegation of keygeneration mechanism required within the enterprise itself. CPABE provides only the user level delegation, but does not provide the full delegation. Moreover, large scale enterprises require scalable revocation mechanism. Li et al. (2011b) proposed the new technique called Hierarchical Attribute Based Encryption (HABE) based on CPABE, which overcomes the CPABE issue. In this scheme, universal attributes are classified into tree structure according to the relationship defined in the access policy. An ancestral node can derive the key of descendant's node and every node is associated with an attribute. Here the user can decrypt the ciphertext only if the number of user's attributes that cover the attributes included in the ciphertext is not less than a threshold value (*d*). The threshold value (*d*) must be set in initial setup phase itself. The major difference between CPABE and HABE is that attributes have the hierarchical structure in HABE but not in CPABE. HABE contains four algorithms:

Setup: Let $U = \{\sigma_{10}, \ldots \sigma_{n0}\}$ be the root attribute set and the depth of the tree in consider as l_i, where i value between 1 and n. The authority generates the public parameter Public Key (PK) and Master Secret Key (MSK) using non-zero random value g_2, $u'_1, \ldots u'_n$, $u_1, \ldots u_l$ from group G. Let α from Z_p.

$$PK = \left(g, g_1, g_2, \widehat{\ell}, (u'_i)_{1 \le i \le n}, (u_i)_{1 \le i \le l}; g_1 = g^{\alpha}\right)$$
$$\text{MSK} = \alpha \quad (20)$$

KeyGeneration(U, PK, MSK): This algorithm generates the secret key of U using PK, MSK, and Access Structure. Let $q(0) = \alpha$. Let k be the

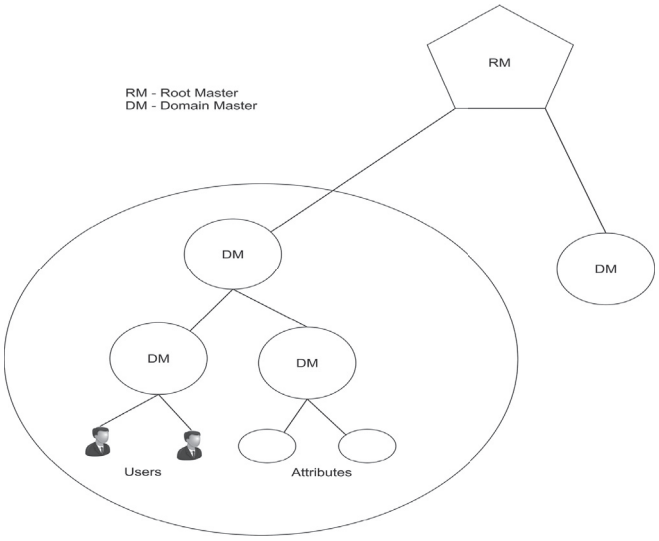

Fig. 12. Architecture of hierarchical attribute based encryption scheme.

depth of the tree. (σ_{i0}, σ_{i1}, · · ·, $\sigma_{i,k-1}$, σ) is the path for σ, for each σ ∈ U. Compute $D_\sigma = (d_{i0}, d_i, d_{i,ki+1}, \cdots, d_{il_i})$, where $d_{i0} = g_2^{g(H(\sigma))} (u'_i \prod_{j=1}^{k} u_{j=1}^{\sigma_{ij}})^r$, $d_i = g^r$, $d_{i,k+1} = u_{k+1}^r, \cdots, d_{il_i} = u_{l_i}^r$; The private key defined as:

$$d_U = \{D_\sigma\} \sigma \in U \tag{21}$$

Encryption(M, U′, PK): The message (M) is encrypted using PK and set of attributes (U′) and it produces the output ciphertext C. Let k' be the depth of the tree. (σ_{j0}, σ'_{j1}, · · ·, $\sigma'_{j,k'-1}$, σ′) is the path for σ′, for each σ' ∈ U'. Choose non-zero random value s in Z_p. Compute $E' = m\, \widehat{e}\, (g_1, g_2)^s$ and $T = g^s$ and compute $E_{\sigma'} = (u'_j \prod_{\delta=1}^{k1} u_\delta^{\sigma'_{j\delta}})^s$ for each σ' ∈ U'. The encrypted text C is defined as:

$$C = (E', T, \{E_{\sigma'}\}) \text{for all } \sigma' \in U' \tag{22}$$

Decryption(C, U′, PK, U, d_U): Data user decrypts the message from ciphertext using secret key d_U. Decryption is only possible if the $|U \cap U'| \geq d$. Choose d-element subset UA in U. (σ_{i0}, σ_{i1}, · · ·, $\sigma_{,k-1}$, σ) is the path for each σ in UA. Similarly σ′ is the attribute in U′ covered by σ with path from the same root σ_{i0} as (σ_{i0}, σ'_{i1}, · · ·, $\sigma'_{j,k'-1}$, σ′). Compute $\sigma_{i\delta} = \sigma'_{i\delta}$ for $1 \leq \delta \leq j$ and $d'_{i0} = d_{i0}\, d_{i,j+1}^{\sigma'_{i,j+1}} \ldots d_{i,j'}^{\sigma'_{i,j'}}$ and message can be decrypted as

$$M = E' \Big/ \prod_{\sigma \in UA} \left(\frac{\widehat{e}(d'_{i0}, T)}{\widehat{e}(d_i, E_{\sigma'})} \right)^{\Delta_{H(\sigma),s(0)}} \tag{23}$$

This scheme used the threshold method, which cannot provide the fine-grained access control. Wang et al. (2010) and their further work (Wang et al., 2011) proposed the full delegation HABE for fine grained access control by combining Hierarchical Identity Based Encryption (HIBE) and CPABE. The architecture of this scheme is shown in Fig. 12. This scheme was also supported the scalable revocation mechanism. This scheme consists of root master and multiple domains. Each domain consists of domain a master who is responsible to create a key for the next level domain masters, and distributing it to the users. Root master is generating keys for the domain masters. Each domain master and attributes are assigned to the unique identifier and each user is assigned with both unique identifier and set of descriptive attributes. Each public key holds the public key with its position in the structure. For example, the public key of DMi with IDi is denoted by (PK_{i-1}, IDi). The problem with this scheme is that the same attributes may be maintained by different domain masters which creates more complexity, and the scheme was not supporting the compound attributes.

Wan et al. (2012) proposed Hierarchical Attribute Set Based Encryption (HASBE) scheme, which supports the scalability, fine-grained access control, and user revocation. It extends the attribute set based encryption with the hierarchical structure, and system model is shown in Fig. 13. Here, root authority is the top-level authority who manages the domain authority. Parent level domain authorities manage each child level domain authorities in hierarchy or data owner, users in their domain. In this scheme, the data owner or data consumers need not be online always. They are only required to be online when needed, but authorities should be always online at all times. The schemes (Wang et al., 2011; Wan et al., 2012) were provided the formal security proof under random oracle model. Moreover, these schemes used conjunctive and disjunctive normal forms to represent the access structure which has the limited expressiveness. Liu et al. (2014) proposed the ciphertext-policy HABE for fine-grained access control and proved that the scheme is secure under q-parallel bilinear Diffie-Hellman exponent assumption. Deng et al. (2014) proposed the new HABE scheme in which the attributes are organized in the matrix. The lower level user can get their access rights from the user has higher level attributes. This concept enables the delegating capability, so it is suitable for large organizations. This scheme was proved under standard model under non-interactive assumptions. In the schemes (Wang et al., 2010, 2011; Wan et al., 2012; Deng et al., 2014), the parent node creates the key for the child node, so that the key generation task is delegated and simplified, but each

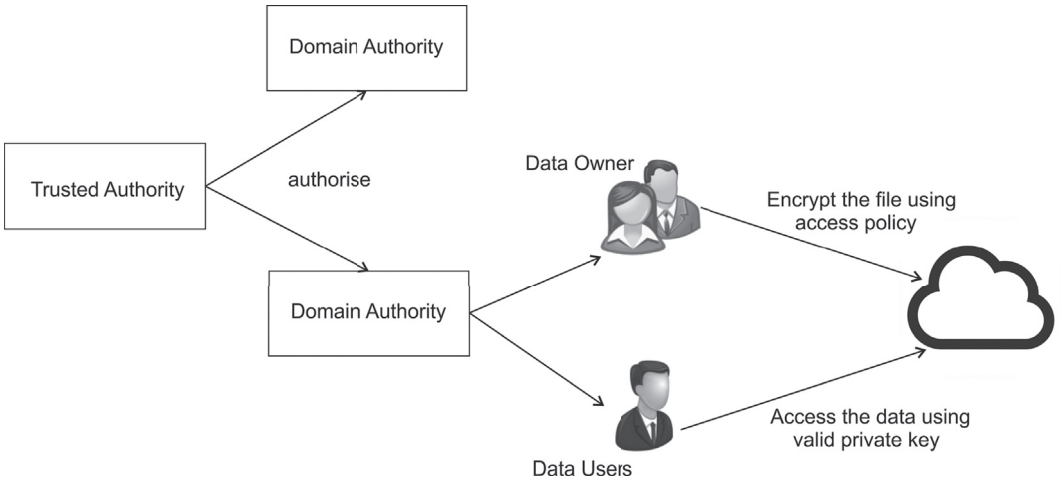

Fig. 13. System model of hierarchical attribute set based encryption scheme.

domain authorities followed the different access structures. Wang et al. (2016) proposed the file-HABE method which was a layered model of access policy. The files are encrypted with one integrated access structure, and this scheme was proved under DBDH assumption.

2.2.5. Big data with CPABE

Big data is a huge volume of complex data and it is growing exponentially with time. The big data may be structured, semi-structured, or unstructured. The structured data means that any data which follows some fixed format like our traditional relational database. The semi-structured data is structured data, but does not follow any formal data model. The unstructured data are a heterogeneous data either it may be text, audio, video, images etc. The major characteristics of big data are volume, variety, velocity and variability.

Big data requires effective storing and processing for the growth of the organization, but the traditional data system like RDBMS is not capable of accommodating big data. Being said that the cloud computing is a better choice to store and process the big data because of the paradigm shift in providing computing resources and data storage. However, when the big data is outsourced to cloud, it suffers with privacy, access control and data duplication challenges. The cloud service alliance (Cloud Security Alliance, 2013) stated that Identity Based Encryption (IBE) and ABE are the two cryptographic techniques that provide effective access control for big data. The ABE is better than IBE because, the IBE requires prior information of data users, but that is not the case in ABE. Henceforth the ABE is a recommend access control mechanism for big data in the cloud (Cavoukian et al., 2015; Strohbach et al., 2016) due to its fine-grained access control nature. In particular CP-ABE is more appropriate to address the big data access control issues because the data owners have the rights to control their data.

However, there are two major issues of the CP-ABE for big data in the cloud addressed so far such as attribute revocation, and policy updating issue. Xiao et al. (2015) proposed the multi authority based efficient access control for big data in which the computation overhead of the data user is reduced by delegating the major part of the decryption process to cloud server. It has the capacity to handle the attributes revocation in which the authorization certificate and user global public key, secret keys are used to handle the attribute revocation instead of the ciphertext re-encryption or update. The security of this method is proved under the generic group model. The problem with this scheme is that whenever the attributes revoked, the authority generates a new certificate, global public and secret key again similar to how if a new user joined in the system and the decryption process mainly depends on the untrusted cloud server and it is also not addressed the user revocation.

The policy update is a big challenge in CPABE for big data in the cloud. Because of a dynamic nature, there is a requirement to change the access policy. Whenever a policy update required first the data owner first retrieves the ciphertext from the cloud, updates the policy, modifies the ciphertext according to the new policy, and then upload it back to the cloud. This process incurs heavy communication cost and computation cost of the data owner. Yang et al. (2015) addressed the policy update problem by outsourcing the update task to cloud server. The update key is generated according to the policy changes and sends the update key to CSP, the CSP updates the same. The security of this scheme is proved under the generic group model. The problem with this scheme is that ciphertext update depends on untrusted CSP and security is proved under generic model. Fugkeaw and Sato (2017) proposed an efficient method of collaborative ciphertext policy attribute role-based encryption for the policy update problem and the security of this scheme is proved against the standard security model. They introduced the role-based access control to identify the read or write privilege of the role, but they never explained how to handle the write access privilege of the user. In both the schemes (Xiao et al., 2015; Fugkeaw and Sato, 2017), the access policy sends along with ciphertext is open to those who access the ciphertext, that leads weak policy privacy. Yang et al. (2017) proposed the scheme that dealt with how to hide the access policy in ciphertext to provide the policy privacy.

The one other another important challenge of big data is the storage overhead in the cloud because of data duplication, there is a chance that there could be more than one data owner for some data, but for different purposes from different department. Each owner might be defined their own access policy according to the requirement, then encrypt the data and upload it into the cloud. If the cloud retains many copies of same big data, then it causes more storage overhead. Data deduplication is a technique that is used to eliminate the redundant data and it is mainly used to reduce the storage overhead. The traditional data deduplication technique is not suitable for encrypted data in the cloud which become big challenge for big data storage and processing in the cloud. The CPABE based data deduplication scheme provided the solution for the above said problem.

Cui et al. (2017) proposed an efficient attribute based deduplication of encrypted data in the hybrid cloud in which data owner generates the tag, label and proof for an association between a tag and label. After encryption, the data owner sends the ciphertext along with tag and label to the cloud but does not send the proof along with the ciphertext. After receiving the storage request from the data owner, the private cloud first verifies the proof and check whether the tag is already available in the tag-label table or not. If it is not available, then it adds the tag and label to the table and sends the label and ciphertext to the public cloud. In case the tag is matched with the existing tag, then it checks whether the access policy of new tag is the subset existing ciphertext policy. If so, then simply discard otherwise check whether existing access policy is the subset of new tag access policy. If yes, then new tag access policy is replaced in the existing ciphertext. In case if that too not the subset, then it performs regeneration of ciphertext to yield both the access policies combined together.

Yan et al. (2016) proposed the deduplication on encrypted data in the cloud using CPABE. The data owner signs the ciphertext and sends it to the cloud and the cloud verifies the signature once it is received. If the verification is done, then cloud checks for similar signature that is available existing ciphertext from the same owner of the data. If a different user stored the same data, then CSP sends the request to the previous data owner by sending the signature. If the data owners approved the signature, then the new ciphertext signature is added with the existing ciphertext. In this scheme, if the signature is disclosed then the system becomes insecure.

3. Comparison

In this section, we compare the various existing ABE schemes in terms of features analysis, security analysis, and performance analysis.

3.1. Feature analysis

Here, we compare different ABE schemes with advantages and disadvantages and it is shown in Table 3. In Table 4, we have given the functionalities of different types of ABE schemes concerning fine-grained access control, collusion resistant, revocation mechanism (user revocation and attribute revocation), and scalability.

3.2. Security analysis

Here, we analyze the security proof of different ABE schemes concerning security model and assumption, and it is tabulated in Table 5.

3.3. Performance analysis

Here, we measured the performance of different ABE schemes based on storage cost, communication cost and computation cost. The storage cost incurs because of public key and secret key in the ABE because data owner stores the public key for encryption and data user stores the secret

Table 3
Comparison of ABE schemes.

Scheme	KP/CP	Access Structure	Advantages	Disadvantages
Sahai and Waters, 2005	Fuzzy IBE	Monotonic	One too many cryptographic public key based encryption and fine-grained access control.	The threshold value used, which is not so expressive and computation overhead is high.
Goyal et al., 2006	KPABE	Monotonic	Tree access structure defined on user private key which helps to improve the computational complexity than Fuzzy IBE.	Does not represent the negative constrains
Ostrovsky et al., 2007		Non-monotonic	Access structure that includes the negative attribute.	More computational overhead
Lewko et al., 2010		Non-monotonic	Achieved the user revocation with small size private keys.	The size of the ciphertext is increased linearly which depends on a number of attributes.
Attrapadung et al., 2011		Non-monotonic	Achieved the constant size ciphertext using identity based on revocation mechanism.	The size of private key is quadratic which is about the number of attributes
Wang and Luo, 2012		Monotonic	Achieved the constant size ciphertext using delerablee identity based broadcast encryption.	The size of private key is quadratic which is about the number of attributes
Lai et al., 2014		Monotonic	Achieved the fully secure constant size ciphertext and fast encryption	The size of private key is quadratic about the number of attributes
Bethencourt et al., 2007	CPABE	Monotonic	Access structure defines on the message which performs better.	Security proved under generic group model
Goyal et al., 2008		Monotonic	Introduced the bounded tree access structure which supports any access formulas.	The depth of the access tree structure is bounded, and it must be specified in the setup algorithm itself.
Emura et al., 2010		Monotonic	Achieved the constant ciphertext length and constant number of bilinear pairing operations.	It was less expressive and used only AND-gate.
Li et al., 2017		Non-monotonic	Improved the performance and efficiency by introducing new access structure ordered binary decision diagram.	Does not support revocation mechanism.
Lewko and Waters, 2011		Monotonic	Proposed the decentralized multi-authority CPABE.	User's attributes might be found by tracking his/her global identifier
Li et al., 2011a		Monotonic	Proposed the multi-authority based user accountability scheme.	Authority only generating keys of all users, so it may decrypt all data.
Nishide et al., 2008	CPABE	Monotonic	Proposed the hidden policy scheme to maintain the policy privacy	This model proved under selectively secure
Phuong et al., 2016		Monotonic	Proposed the hidden policy scheme with constant size ciphertext	This model proved under selectively secure
Xu et al., 2016		Monotonic	Reduced the computational cost of the private key and handle user revocation	The central authority may decrypt all the data because CA only issuing all private keys to users.
Wang et al., 2011	HABE	Monotonic	Providing full delegation and supports for scalability	Does not support compound attributes and same attributes might be repeated in different domain.
Wan et al., 2012		Monotonic	Supporting compound attributes and support for scalability	Limited expressive because access structure used CNF and DNF formulas.
Wang et al., 2016		Monotonic	One integrated access structure used. Reduce the computational complexity and storage cost.	It does not support revocation mechanism.

Table 4
Comparison of various ABE Scheme functionalities.

Scheme	Fine-grained access control	Collusion resistant	Revocation mechanism	Scalability
FIBE	Yes	Yes	No	No
KPABE	Yes	Yes	Yes	No
CPABE	Yes	Yes	Yes	No
HABE	Yes	Yes	Yes (User)	Yes

Table 5
Comparison of various ABE schemes security model.

Scheme	KP/CP ABE	Security model	Security assumption
Goyal et al., 2006	KPABE	Selective	DBDH
Ostrovsky et al., 2007		Selective	DBDH
Lewko et al., 2010		Adaptive	DBDH & DLIN
Lai et al., 2014		Fully	DBDH
Chase, 2007		Adaptive	BDH
Han et al., 2012		Selective	DBDH
Bethencourt et al., 2007	CPABE	Fully	Generic Group
Emura et al., 2010		Selective	DBDH
Waters, 2011		Selective	DPBDHE
Lewko and Waters, 2011		Fully	Subgroup
Li et al., 2011a		Selective	DBDH & q-DDH
Nishide et al., 2008		Selective	DBDH & DLIN
Lai et al., 2011		Fully	Subgroup
Li et al., 2012		Fully	DBDH
Liang et al., 2009b		Selective	ADBDH
Liang et al., 2013		Selective	D q-parallel BDHE
Xu et al., 2016		Selective	DBDH
Liang et al., 2010		Selective	DBDH
Yu et al., 2010		Selective	DBDH
Hur and Noh, 2011		Forward & Backward	BDH
Yang et al., 2013		Forward & Backward	q-parallel BDHE
Wang et al., 2011		Adaptive	BDH
Wan et al., 2012		Fully	Generic Group

Table 6
Notations used in performance analysis.

Notations	Meaning
η_{UA}	number of universal attributes
η_u	number of users
η_∂	number of user attributes
η_l	number attributes in the ciphertext
η_{nln}	number of non-leaf nodes in the access tree
$\eta_{\partial p}$	number of user attributes in the user access policy
η_{au}	number of authorities
η_{vp}	number of valid paths in ordered binary decision diagram
$Ł_S$	length of the element in group G_S
$Ł_T$	length of the element in group G_T
t_e	time required for one exponentiation operation
t_p	time required for one pairing operation

key for decryption. The major communication overhead occurs because of uploading and downloading the ciphertext, so ciphertext size determines the communication cost of the system. The major computation cost occurs because of data encryption and data decryption by data owner and data users respectively. Thus the overall performance of the ABE systems is determined based on size of public key (data owner storage cost), secret key (data user storage cost), ciphertext (communication cost), and encryption cost (data owner computation cost), decryption cost (data user computation cost). The various notations used in the performance analysis are shown in Table 6. The different ABE scheme's storage cost and communication cost computed and it is tabulated in Table 7 and their computation overhead shown in Table 8.

4. Applications of ABE

In this section, we give the application scenario of different ABE schemes such as KPABE, and CPABE.

In KPABE, the data is encrypted using set of descriptive attributes and secret key is associated with access structure. KPABE is more suitable for the following application which requires the descriptive attribute to encrypt the data such as 1) In secure forensic application, the relevant information and evidence is stored with set of descriptive attributes such as crime id, date, and name of the person, etc. and this could be accessed only by analyst 2) In network audit log application, the log is stored with the identity of machine ipaddress, user name, date and time etc. The admin can access the required log records based on the requirement. In both the scenario, the data need to be encrypted based on described attributes. However, the KPABE is not suitable for application in which the data owner needs to control the data because the problem with KPABE is the data owner does not have the rights to control the data.

In CPABE, the data is encrypted under defined access structure and secret key is associated with set of user attributes. The CPABE suits well in the following application scenario 1) In collaborative project development, it allows only the people involved in the project to access the data 2) Video surveillance system is widely used to ensure the security of the organization, home and it also used to identify people violating the traffic rules etc. The accessing of the video data is restricted to designated people like security officers, higher grade officers etc. in an organization because of privacy 3) The employees of an organization could only access their department data not entire data of the organization. For example, the production department employee can access only the production department data and is restricted to access sales, and accounting data etc. while the production department data is accessible to general manager, CEO and MD 4) In University, the professor teaching the cloud computing subject can share the course content, related assignments, and class schedules to the students who have only opted the cloud computing subject and after the course completion the student's access is not permitted. Since the professor knows the time period the students need to access to the materials and information the time based rekeying CPABE revocation mechanism is more apt for this application. However the problem with CPABE is that it is not supporting scalability, so CPABE may not be suitable for large enterprise which requires scalability.

In order to cope up with the large enterprise requirement HABE was proposed which is CPABE along with hierarchical property of the enterprise. HABE supports scalability by delegating the key generation task

Table 8
Computation cost performance of existing ABE schemes.

Scheme	KP/CP/HABE	Access structure	Computation cost	
			Encryption cost	Decryption cost
Goyal et al., 2006	KPABE	Tree	$(\eta_l)\ t_e + t_p$	$(\eta_{\partial p} + 1)\ t_p + (\eta_{nln})\ t_e$
Ostrovsky et al., 2007		LSSS	$(2\eta_l + 1)\ t_e + t_p$	$(5\eta_{\partial p})\ t_p + (2\eta_{\partial p})\ t_e$
Han et al., 2012		Tree	$(\eta_l + 2)\ t_e + t_p$	$(\eta_{au} + \eta_{\partial p} + 1)\ t_p + (\eta_{\partial p})\ t_e$
Bethencourt et al., 2007	CPABE	Tree	$(2\eta_l + 1)\ t_e + t_p$	$(2\eta_\partial + 1)\ t_p + (\eta_{nln})\ t_e$
Ibraimi et al., 2009a		Tree	$(\eta_l)\ t_e + t_p$	$(\eta_\partial + 1)\ t_p + (\eta_{nln})\ t_e$
Li et al., 2017		OBDD	$(\eta_{vp} + 1)\ t_e + t_p$	$2t_p$
Lewko and Waters, 2011		LSSS	$(2\eta_l)\ t_e + (2\eta_l + 1)\ t_p$	$(2\eta_\partial)\ t_p + t_e$
Hur and Noh, 2011		Tree	$(2\eta_l + 1)\ t_e + t_p$	$(2\eta_\partial + 1)\ t_p + (\eta_{nln})\ t_e + (\log \eta_u)$
Yang and Jia, 2014		LSSS	$(4\eta_l + 2)\ t_e + t_p$	$(2\eta_\partial + 4)\ t_p$
Chen and Ma, 2014		LSSS	$(\eta_l + \eta_{au} + 1)\ t_e + t_p$	$(\eta_{au} + 2\eta_\partial + 1)\ t_p + (\eta_\partial)\ t_e$
Xiao et al., 2015		LSSS	$(3\eta_l)\ t_e + (2\eta_l + 1)\ t_p$	t_e
Deng et al., 2014	HABE	LSSS	$(4\eta_l + 1)\ t_e + t_p$	$(3\eta_\partial + 1)\ t_p + (\eta_\partial)\ t_e$

Table 7
Storage cost and communication cost performance of existing ABE schemes.

Scheme	KP/CP/HABE	Access structure	Storage cost		Communication cost
			Public key size	Secret key size	Ciphertext size
Goyal et al., 2006	KPABE	Tree	$(\eta_{UA})\ Ł_S + Ł_T$	$(\eta_{\partial p})\ Ł_S$	$(\eta_l)\ Ł_S + Ł_T$
Ostrovsky et al., 2007		LSSS	$(2\eta_l+2)\ Ł_S$	$(5\eta_{\partial p})\ Ł_S$	$(2\eta_l + 1)\ Ł_S + Ł_T$
Han et al., 2012		Tree	$(\eta_{UA} + 1)\ Ł_S + Ł_T$	$(\eta_{\partial p} + 1)\ Ł_S$	$(\eta_l + 2)\ Ł_S + Ł_T$
Bethencourt et al., 2007	CPABE	Tree	$Ł_S + Ł_T$	$(\eta_\partial + 1)\ Ł_S$	$(2\eta_l + 1)\ Ł_S + Ł_T$
Ibraimi et al., 2009a		Tree	$(\eta_{UA})\ Ł_S + Ł_T$	$(\eta_\partial + 1)\ Ł_S$	$(\eta_l + 1)\ Ł_S + Ł_T$
Li et al., 2017		OBDD	$(2\eta_{UA})\ Ł_S + Ł_T$	$2\ Ł_S$	$(\eta_{vp} + 1)\ Ł_S + Ł_T$
Lewko and Waters, 2011		LSSS	$(\eta_{UA})\ Ł_S + Ł_T$	$(\eta_\partial)\ Ł_S$	$(2\eta_l)\ Ł_S + (\eta_l + 1)\ Ł_T$
Hur and Noh, 2011		Tree	$Ł_S + Ł_T$	$(2\eta_\partial + 1)\ Ł_S + (\log \eta_u)$	$(2\eta_l + 1)\ Ł_S + Ł_T$
Yang and Jia, 2014		LSSS	$(2\eta_{UA} + 4)\ Ł_S$	$(\eta_\partial + 2)\ Ł_S$	$(4\eta_l + 2)\ Ł_S + Ł_T$
Chen and Ma, 2014		LSSS	$(2\eta_{UA})\ Ł_S + Ł_T$	$(\eta_\partial + 1)\ Ł_S$	$(\eta_l + \eta_{au} + 1)\ Ł_S + Ł_T$
Xiao et al., 2015		LSSS	$(2\eta_{UA})\ Ł_S + (\eta_{UA})\ Ł_T$	$(\eta_\partial)\ Ł_S$	$(2\eta_l)\ Ł_S + (\eta_l + 1)\ Ł_T$
Deng et al., 2014	HABE	LSSS	$(2\eta_{UA} + 1)\ Ł_S + Ł_T$	$(\eta_\partial + 2)\ Ł_S$	$(3\eta_l + 1)\ Ł_S + Ł_T$

to next level authority. HABE is more suitable for large enterprise and the application requires hierarchal property or scalability. 1) In personal health information system, the patient shares his/her medial record only to concern doctor, concern department nurse, and insurance people. Suppose if the doctor wants to consult with another doctor, or expert from other hospital, then the doctor will authorize the control to others 2) In University a department organises the conference, the head of the department assigns the paper review process to a particular senior faculty and she will take care of the entire review process with the help of review committee faculty members. Here, the head giving access rights to senior faculty, senior faulty giving rights to other faulty members. In both above said applications HABE is more suitable which holds the hierarchy property. However, HABE may not be suitable for application which needs to address the attribute revocation.

5. Future directions

We analysed the different ABE schemes and research progress. It is a leading research area in the past decade but still there is a scope to investigate further on ABE. The following are the some of the possible challenges for further investigation on ABE.

1 *CPABE efficiency for big data, mobile, IoT applications:* CPABE scheme provides effective access control for big data, mobile and IoT applications in the cloud but the computation requirement of encryption and decryption process is inefficient on big data due to huge volume and on mobile, IoT devices due to physical limitations of storage and processing capability. CPABE in big data, mobile and IoT applications is an emerging research field and the following are the some of the open challenges:
   a. Maximum ABE schemes were proposed based on bilinear pairings theory. It requires more number of paring which is an expensive operation. There is a scope to reduce overhead of paring based ABE scheme by the following ways 1) reduce the number of pairing operation in decryption process 2) design the short ciphertext, and short secret key 3) outsource the partial encryption and decryption process to CSP or proxy without compromising the security 4) the alternate method to bilinear paring based ABE is to construct the ABE scheme using lattice harness problem (Rahman et al., 2017), Elliptic Curve Cryptography (ECC) (Odelu and Das, 2016) or pairing based multi-base number representation technique (MBNR) (Chandrasekaran and Balakrishnan, 2016).
   b. The length of the ciphertext and secret key is proportional to the number of attributes in the access policy and the user attributes respectively. The length of ciphertext and secret key affects the performance of encryption and decryption, so there is a scope to design constant ciphertext and constant secret key to improve the efficiency of ABE.
   c. The policy update process incurs high communication and computation overhead. How to effectively outsource the policy update process to CSP or proxy is still challenging open issue.
2 *Efficient revocation mechanism:* The practical applications need to support both the attribute revocation and user revocation. There are two fundamental security issues of revocation which are forward and backward security. Forward security means the revoked users should not access the subsequently published ciphertext using old secret key. Backward security means the newly joined user should give the privilege to access previously published ciphertext if the new user attribute set satisfy the access policy of the ciphertext. Moreover, during revocation the data owner is required to update the ciphertext and data users are required to update their secret key. This process incurs the computation overhead for data owner and data users. Majority of the existing ABE schemes addressed either attribute revocation or user revocation and it prevents either forward or backward security. Henceforth there is a scope to address the revocation challenges by the following way 1) develop ABE scheme which will support both user revocation and attribute revocation without compromising the forward and backward security 2) delegate the ciphertext and secret key update process to CSP or proxy without compromising the security. Along with that also address the open challenge of attribute revocation in HABE.
3 *Ordered binary decision diagram:* Recently CPABE scheme with ordered binary decision diagram access structure was introduced. The user revocation, attribute revocation, policy updating features is not addressed so far in OBDD access structure, so there is a scope to address the revocation mechanism, policy updating features on ordered binary decision diagram access structure.
4 *Accountability:* The key abusing and user colluding key attack are the two major issues in ABE. The system must monitor the authorities and users to find out the malicious user who performs key abusing and key colluding attack. Thus it is necessary to develop high effective accountable system in ABE.
5 *Practical application:* Only a very few ABE schemes approached for practical application scenario like keyword search on encrypted data and personal health record (Pussewalage and Oleshchuk, 2016). In future there is a variety of scope available to address applicability of ABE schemes in the practical applications.

The ABE open challenges mainly focus on three major aspects such as efficiency, revocation, and monitoring key abuse. The major constraints of efficiency is storage, communication, and computation overhead because of size of the public key, secret key, size of ciphertext, encryption, decryption time, and ciphertext update time. The future will work focus on reducing the various overhead of ABE by the following 1) designing short or constant size keys and ciphertext 2) outsourcing the heavy computation task like encryption, decryption, ciphertext, secret key or policy update task 3) developing ABE using lattices, ECC, MBNR methods. The ABE scheme must effectively handle both user revocation and attribute revocation without compromising the forward and backward security. The final aspect of ABE scheme is accountability in which the system should monitor the user who abuses the key to identify the malicious user.

6. Discussion and conclusion

In this paper, we focused on the importance and requirement of privacy and access control in the cloud environment and ABE is a widely used prominent cryptographic technique to provide that privacy and the fine-grained access control. We comprehensively surveyed the attribute based encryption schemes based on various characteristics and parameters such as access structure, multi-authority, hidden policy in CPABE, proxy re-encryption in CPABE, revocation mechanism in CPABE and HABE. Furthermore, we analysed the advantages, disadvantages, functionalities, security model, security assumptions and efficiency of different ABE schemes. Some of the observations and findings of different ABE schemes are as follows.

- CPABE performs better than KPABE by giving full control to data owner on their data.
- Maximum ABE algorithms were proposed based on bilinear map with variant of decisional bilinear Diffie-Hellman assumption. The most ABE schemes are selectively secure under either chosen-plaintext attack or chosen-ciphertext attack. The general method followed to prove the security is game model between challenger and adversary.
- Most commonly used access structures in CPABE schemes are tree and LSSS matrix structures. From Tables 6 and 7, it is clearly noticed that most of the LSSS matrix access structure required more computation time for encryption because of ciphertext size but it provides better decryption cost. The ABE schemes with tree access structure produces better encryption cost, but it requires more decryption cost than the LSSS matrix access structure.

- The most recently introduced CPABE with ordered binary decision diagram access structure performs better than both tree and LSSS matrix access structure in terms of size of secret key, ciphertext, encryption, and decryption cost.
- The CPABE schemes addressed the either user revocation or attribute revocation. Only few algorithms addressed both user and attribute revocation. In attribute revocation, the schemes only addressed the problem of updating new attributes, but did not address the addition and deletion of attributes.
- The update keying method of CPABE revocation schemes performs better than proxy re-encryption and time based rekeying method because it only updates the required portion of the ciphertext where as proxy re-encryption schemes entirely re-encrypt the ciphertext which increase the communication and computation cost and time based re-keying method revocation schemes only addressed the user revocation.

In addition to that, we also identified the suitability and unsuitability of different types of ABE methods for practical applications. Finally, we conclude the survey with some open challenges that needs to be investigated further such as efficiency and security improvements required in CPABE schemes for big data, mobile and IoT applications, efficient way of handling revocation mechanism, accountability and develop the practical application based on ABE.

References

Attrapadung, N., Imai, H., 2009. Attribute-based encryption supporting direct/indirect revocation modes. In: Cryptography and Coding. Springer, Berlin Heidelberg, pp. 278–300. https://doi.org/10.1007/978-3-642-10868-6_17.

Attrapadung, N., Libert, B., De Panafieu, E., 2011. Expressive key-policy attribute-based encryption with constant-size ciphertexts. In: Public Key Cryptography. Springer, Berlin Heidelberg, pp. 90–108. https://doi.org/10.1007/978-3-642-19379-8_6.

Bethencourt, J., Sahai, A., Waters, B., 2007. Ciphertext-policy attribute-based encryption. In: IEEE Symposium on Security and Privacy. IEEE, pp. 321–334. https://doi.org/10.1109/SP.2007.11.

Cavoukian, A., Chibba, M., Williamson, G., Ferguson, A., 2015. The Importance of ABAC: Attribute-based Access Control to Big Data: Privacy and Context. Privacy and Big Data Institute, Ryerson University, Toronto, Canada.

Chandrasekaran, B., Balakrishnan, R., 2016. Efficient pairing computation for attribute based encryption using MBNR for big data in cloud. In: 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT). IEEE, pp. 243–247. http://doi.org/10.1109/ICATCCT.2016.7912001.

Chase, M., 2007. Multi-authority attribute based encryption. In: In Theory of Cryptography Conference, vol. 4392, pp. 515–534. Berlin Heidelberg. https://doi.org/10.1007/978-3-540-70936-7_28.

Chase, M., Chow, S.S., 2009. Improving privacy and security in multi-authority attribute-based encryption. In: Proceedings of 16th ACM Conference Computer and Communications Security. ACM, pp. 121–130. https://doi.org/10.1145/1653662.1653678.

Chen, J., Ma, H., 2014. Efficient decentralized attribute-based access control for cloud storage with user revocation. In: Proceedings of International Conference on Communication. IEEE, pp. 3782–3787. https://doi.org/10.1109/ICC.2014.6883910.

Cheung, L., Newport, C.C., 2007. Provably secure ciphertext policy abe. In: ACM Conference on Computer and Communications Security. ACM, pp. 456–465. https://doi.org/10.1145/1315245.1315302.

Chow, S.S.M., 2016. A framework of multi-authority attribute-based encryption with outsourcing and revocation. In: Proceedings of 21st ACM Symposium on Access Control Models Technologies. ACM, pp. 215–226. http://doi.org/10.1145/2914642.2914659.

Cloud Security Alliance, 2013. Expanded Top Ten Big Data Security and Privacy Challenges.

Cui, H., Deng, R.H., Li, Y., Wu, G., 2017. Attribute-based storage supporting secure deduplication of encrypted data in cloud. IEEE Trans. Big Data. PP(99). https://doi.org/10.1109/TBDATA.2017.2656120.

Delerablee, C., 2007. Identity-based broadcast encryption with constant size ciphertexts and private keys. In: Theory and Application of Cryptology and Information Security. Springer, Berlin Heidelberg, pp. 200–215. https://doi.org/10.1007/978-3-540-76900-2_12.

Deng, H., Wu, Q., Qin, B., Domingo-Ferrer, J., Zhang, L., Liu, J., Shi, W., 2014. Ciphertext-policy hierarchical attribute-based encryption with short ciphertexts. Inf. Sci. 275, 370–384. http://doi.org/10.1016/j.ins.2014.01.035.

Emura, K., Miyaji, A., Omote, K., Nomura, A., Soshi, M., 2010. A ciphertext-policy attribute-based encryption scheme with constant ciphertext length. Int. J. Appl. Cryptogr. 2 (1), 46–59. https://doi.org/10.1504/IJACT.2010.033798.

Fan, C.I., Huang, V.S.M., Ruan, H.M., 2014. Arbitrary-state attribute based encryption with dynamic membership. IEEE Trans. Comput. 63 (8), 1951–1961. https://doi.org/10.1109/TC.2013.83.

Fehling, C., Leymann, F., Retter, R., Schupeck, W., Arbitter, P., 2014. Cloud computing fundamentals. In: Cloud Computing Patterns. Springer, Vienna, pp. 21–78. https://doi.org/10.1007/978-3-7091-1568-8_2.

Fugkeaw, S., Sato, H., 2017. Scalable and secure access control policy update for outsourced big data. Future Generat. Comput. Syst. 79, 364–373. https://doi.org/10.1016/j.future.2017.06.014.

Goyal, V., Pandey, O., Sahai, A., Waters, B., 2006. Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security. ACM, pp. 89–98. https://doi.org/10.1145/1180405.1180418.

Goyal, V., Jain, A., Pandey, O., Sahai, A., 2008. Bounded ciphertext policy attribute based encryption. In: Automata, Languages and Programming, pp. 579–591. https://doi.org/10.1007/978-3-540-70583-3_47.

Han, J., Susilo, W., Mu, Y., Yan, J., 2012. Privacy-preserving decentralized key-policy attribute-based encryption. IEEE Trans. Parallel Distr. Syst. 23 (11), 2150–2162. https://doi.org/10.1109/TPDS.2012.50.

He, H., Li, R., Dong, X., Zhang, Z., 2014. Secure, efficient and fine-grained data access control mechanism for P2P storage cloud. IEEE Trans. Cloud Comput. 2 (4), 471–484. https://doi.org/10.1109/TCC.2014.2378788.

Hur, J., 2013. Improving security and efficiency in attribute-based data sharing. IEEE Trans. Knowl. Data Eng. 25 (10), 2271–2282. https://doi.org/10.1109/TKDE.2011.78.

Hur, J., Noh, D.K., 2011. Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Trans. Parallel Distr. Syst. 22 (7), 1214–1221. https://doi.org/10.1109/TPDS.2010.203.

Ibraimi, L., Hartel, Q.P., Jonker, W., 2009. Efficient and provable secure ciphertext-policy attribute-based encryption schemes. In: Information Security Practice and Experience. Springer, Berlin Heidelberg, pp. 1–12. https://doi.org/10.1007/978-3-642-00843-6_1.

Ibraimi, L., Petkovic, M., Nikova, S.I., Hartel, P.H., Jonker, W., 2009. Mediated ciphertext-policy attribute-based encryption and its application. In: Information Security Applications. Springer, Berlin Heidelberg, pp. 309–323. https://doi.org/10.1007/978-3-642-10838-9_23.

Jadeja, Y., Modi, K., 2012. Cloud computing-concepts, architecture and challenges. In: International Conference on Computing. Electronics and Electrical Technologies (ICCEET), IEEE, pp. 877–880. https://doi.org/10.1109/ICCEET.2012.6203873.

Jin, C., Feng, X., Shen, Q., 2016. Fully secure hidden ciphertext policy attribute-based encryption with short ciphertext size. In: Proceedings of the 6th International Conference on Communication and Network Security. ACM, pp. 91–98. https://doi.org/10.1145/3017971.3017981.

Kamara, S., Lauter, K., 2010. Cryptographic cloud storage. In: International Conference on Financial Cryptography and Data Security. Springer, Berlin Heidelberg, pp. 136–149. https://doi.org/10.1007/978-3-642-14992-4_13.

Khan, A.R., 2012. Access control in cloud computing environment. ARPN J. Eng. Appl. Sci. 7 (5), 613–615.

Lai, J., Deng, R.H., Li, Y., 2011. Fully secure cipertext-policy hiding CP-ABE. In: 7th International Conference on Information Security Practice and Experience. Springer, Berlin Heidelberg, pp. 24–39. https://doi.org/10.1007/978-3-642-21031-0_3.

Lai, J., Deng, R.H., Li, Y., Weng, J., 2014. Fully secure key-policy attribute-based encryption with constant-size ciphertexts and fast decryption. In: Proceedings of the 9th ACM Symposium on Information Computer and Communications Security. ACM, pp. 239–248. https://doi.org/10.1145/2590296.2590334.

Lewko, A., Waters, B., 2011. Decentralizing attribute-based encryption. In: Theory and Applications of Cryptographic Techniques, vol. 6632. Springer, Berlin Heidelberg, pp. 568–588. https://doi.org/10.1007/978-3-642-20465-4_31.

Lewko, A., Sahai, A., Waters, B., 2010. Revocation systems with very small private keys. In: Security and Privacy (SP). IEEE, pp. 273–285. https://doi.org/10.1109/SP.2010.23.

Li, K., 2013. Matrix Access Structure Policy Used in Attribute Based Proxy Re-encryption. arXiv preprint. http://arxiv.org/abs/1302.6428.

Li, J., Ren, K., Zhu, B., Wan, Z., 2009. Privacy-aware attribute-based encryption with user accountability. In: Information Security. Springer, Berlin Heidelberg, pp. 347–362. https://doi.org/10.1007/978-3-642-04474-8_28.

Li, J., Huang, Q., Chen, X., Chow, S.S.M., Wong, D.S., Xie, D., 2011. Multi-authority ciphertext-policy attribute-based encryption with accountability. In: Proceedings ACM Symposium on Computer and Communication Security. ACM, pp. 386–390. https://doi.org/10.1145/1966913.1966964.

Li, J., Wang, Q., Wang, C., Ren, K., 2011. Enhancing attribute-based encryption with attribute hierarchy. Mobile Network. Appl. 16 (5), 553–561. https://doi.org/10.1007/s11036-010-0233-y.

Li, X., Gu, D., Ren, Y., Ding, N., Yuan, K., 2012. Efficient ciphertext-policy attribute based encryption with hidden policy. In: International Conference on Internet and Distributed Computing Systems. Springer, Berlin Heidelberg, pp. 146–159. https://doi.org/10.1007/978-3-642-34883-9_12.

Li, W., Xue, K., Xue, Y., Hong, J., 2016. TMACS: a robust and verifiable threshold multi-authority access control system in public cloud storage. IEEE Trans. Parallel Distr. Syst. 27 (5), 1484–1496. https://doi.org/10.1109/TPDS.2015.2448095.

Li, Q., Ma, J., Li, R., Liu, X., Xiong, J., Chen, D., 2016. Secure, efficient and revocable multi-authority access control system in cloud storage. Comput. Secur. 59, 45–59. http://doi.org/10.1016/j.cose.2016.02.002.

Li, L., Gu, T., Chang, L., Xu, Z., Liu, Y., Qian, J., 2017. A ciphertext-policy attribute-based encryption based on an ordered binary decision diagram. IEEE Access 5, 1137–1145. https://doi.org/10.1109/ACCESS.2017.2651904.

Liang, X., Cao, Z., Lin, H., Xing, D., 2009. Provably secure and efficient bounded ciphertext policy attribute based encryption. In: Information, Computer, and Communications Security. ACM, pp. 343–352. https://doi.org/10.1145/1533057.1533102.

Liang, X., Cao, Z., Lin, H., Shao, J., 2009. Attribute based proxy re-encryption with delegating capabilities. In: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security. ACM, pp. 276–286. https://doi.org/10.1145/1533057.1533094.
Liang, X., Lu, R., Lin, X., Shen, X.S., 2010. Ciphertext Policy Attribute Based Encryption with Efficient Revocation, Technical Report. University of Waterloo.
Liang, K., Fang, L., Susilo, W., Wong, D.S., 2013. A ciphertext-policy attribute-based proxy re-encryption with chosen-ciphertext security. In: Proceedings of 5th International Conference on Intelligent Networking and Collaborative Systems. IEEE, pp. 552–559. https://doi.org/10.1109/INCoS.2013.103.
Liang, K., Au, M.H., Liu, J.K., Susilo, W., Wong, D.S., Yang, G., Yu, Y., Yang, A., 2015. A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. Future Generat. Comput. Syst. 52, 95–108. http://doi.org/10.1016/j.future.2014.11.016.
Liu, Z., Cao, Z., Huang, Q., Wong, D.S., Yuen, T.H., 2011. Fully secure multi-authority ciphertext-policy attribute-based encryption without random oracles. In: Computer Security. Springer, Berlin Heidelberg, pp. 278–297. https://doi.org/10.1007/978-3-642-23822-2_16.
Liu, X., Ma, J., Xiong, J., Liu, G., 2014. Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data. IJ Netw. Secur. 16 (6), 437–443. http://doi.org/10.6633/IJNS.201411.16 (6).05.
Lumb, I., Choi, E., Rimal, B.P., 2009. A taxonomy and survey of cloud computing systems. In: International Conference on Networked Computing and Advanced Information Management. IEEE, pp. 44–51. https://doi.org/10.1109/NCM.2009.218.
Luo, S., Hu, J., Chen, Z., Soriano, M., Qing, S., López, J., 2010. Ciphertext policy attribute-based proxy re-encryption. In: Information and Communications Security, vol. 6476. Springer, Berlin Heidelberg, pp. 401–415. https://doi.org/10.1007/978-3-642-17650-0_28.
Nishide, T., Yoneyama, K., Ohta, K., 2008. Attribute-based encryption with partially hidden encryptor-specified access structures. In: International Conference on Applied Cryptography and Network Security. Springer, Berlin Heidelberg, pp. 111–129. https://doi.org/10.1007/978-3-540-68914-0_7.
Odelu, V., Das, A.K., 2016. Design of a new CP-ABE with constant-size secret keys for lightweight devices using elliptic curve cryptography. Secur. Commun. Network. 9 (17), 4048–4059. http://doi.org/10.1002/sec.1587.
Ostrovsky, R., Sahai, A., Waters, B., 2007. Attribute-based encryption with non-monotonic access structures. In: ACM Conference on Computer and Communications Security Computer and Communications Security. ACM, pp. 195–203. https://doi.org/10.1145/1315245.1315270.
Pasupuleti, S.K., Ramalingam, S., Buyya, R., 2016. An efficient and secure privacy-preserving approach for outsourced data of resource constrained mobile devices in cloud computing. J. Netw. Comput. Appl. 64, 12–22. https://doi.org/10.1016/j.jnca.2015.11.023.
Phuong, T.V.X., Yang, G., Susilo, W., 2016. Hidden ciphertext policy attribute-based encryption under standard assumptions. IEEE Trans. Inf. Forensics Secur. 11 (1), 35–45. https://doi.org/10.1109/TIFS.2015.2475723.
Pussewalage, H.S.G., Oleshchuk, V., 2016. A patient-centric attribute based access control scheme for secure sharing of personal health records using cloud computing. In: 2nd International Conference on Collaboration and Internet Computing (CIC). IEEE, pp. 46–53. http://doi.org/10.1109/CIC.2016.020.
Rahman, M.S., Basu, A., Kiyomoto, S., 2017. Decentralized ciphertext-policy attribute-based encryption: a post-quantum construction. J. Int. Serv. Inf. Secur.JISIS 7 (3), 1–16. http://doi.org/10.22667/JISIS.2017.08.31.001.
Rahulamathavan, Y., Veluru, S., Han, J., Li, F., Rajarajan, M., Lu, R., 2016. User collusion avoidance scheme for privacy-preserving decentralized key-policy attribute-based encryption. IEEE Trans. Comput. 65 (9), 2939–2946. https://doi.org/10.1109/TC.2015.2510646.
Sahai, A., Waters, B., 2005. Fuzzy identity-based encryption. In: Theory and Applications of Cryptographic Techniques. Springer, Berlin Heidelberg, pp. 457–473. https://doi.org/10.1007/11426639_27.
Seo, H., Kim, H., 2012. Attribute-based proxy re-encryption with a constant number of pairing operations. J. Inf. Commun. Eng. 10 (1), 53–60. https://doi.org/10.6109/jicce.2012.10.1.053.
Strohbach, M., Daubert, J., Ravkin, H., Lischka, M., 2016. Big data storage. In: New Horizons for a Data-driven Economy. Springer International Publishing, pp. 119–141. https://doi.org/10.1007/978-3-319-21569-3_7.
Takabi, H., Joshi, J.B.D., Ahn, G., 2010. Security and privacy challenges in cloud computing environments. IEEE Secur. Priv. 8 (6), 24–31. https://doi.org/10.1109/MSP.2010.186.
Wan, Z., Liu, J.E., Deng, R.H., 2012. HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Trans. Inf. Forensics Secur. 7 (2), 743–754. https://doi.org/10.1109/TIFS.2011.2172209.
Wang, C.J., Luo, J.F., 2012. A key-policy attribute-based encryption scheme with constant size ciphertext. In: Eighth International Conference on Computational Intelligence and Security (CIS). IEEE, pp. 447–451. https://doi.org/10.1109/CIS.2012.106.
Wang, G., Liu, Q., Wu, J., 2010. Hierarchical attribute-based encryption for fine-grained access control in cloud storage services. In: Proceedings of the 17th ACM Conference on Computer and Communications Security. ACM, pp. 735–737. https://doi.org/10.1145/1866307.1866414.
Wang, G., Liu, Q., Wu, J., Guo, M., 2011. Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers. Comput. Secur. 30 (5), 320–331. http://doi.org/10.1016/j.cose.2011.05.006.
Wang, S., Zhou, J., Liu, J.K., Yu, J., Chen, J., Xie, W., 2016. An efficient file hierarchy attribute-based encryption scheme in cloud computing. IEEE Trans. Inf. Forensics Secur. 11 (6), 1265–1277. https://doi.org/10.1109/TIFS.2016.2523941.
Waters, B., 2011. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. In: Public Key Cryptography. Springer, Berlin Heidelberg, pp. 53–70. https://doi.org/10.1007/978-3-642-19379-8_4.
Wei, J., Liu, W., Hu, X., 2016. Secure and efficient attribute-based access control for multiauthority cloud storage. IEEE Syst. J. 1–12. PP(99). https://doi.org/10.1109/JSYST.2016.2633559.
Wu, Q.X., Zhang, M., 2012. Adaptively secure attribute-based encryption supporting attribute revocation. China Commun. 9 (9), 22–40.
Wu, J., Ping, L., Ge, X., Wang, Y., Fu, J., 2010. Cloud storage as the infrastructure of cloud computing. In: International Conference on Intelligent Computing and Cognitive Informatics (ICICCI). IEEE, pp. 380–383. https://doi.org/10.1109/ICICCI.2010.119.
Xiao, M., Wang, M., Liu, X., Sun, J., 2015. Efficient distributed access control for big data in clouds. In: Computer Communications Workshops (INFOCOM WKSHPS). IEEE, pp. 202–207. https://doi.org/10.1109/INFCOMW.2015.7179385.
Xu, X., Zhou, J., Wang, X., Zhang, Y., 2016. Multi-authority proxy re-encryption based on CPABE for cloud storage systems. J. Syst. Eng. Electron. 27 (1), 211–223.
Yan, Z., Wang, M., Li, Y., Vasilakos, A.V., 2016. Encrypted data management with deduplication in cloud computing. IEEE Trans. on Cloud Comput. 3 (2), 28–35. http://doi.org/10.1109/MCC.2016.29.
Yang, K., Jia, X., 2014. Expressive, efficient, and revocable data access control for multi-authority cloud storage. IEEE Trans. Parallel Distr. Syst. 25 (7), 1735–1744. https://doi.org/10.1109/TPDS.2013.253.
Yang, K., Jia, X., Ren, K., Zhang, B., Xie, R., 2013. DAC-MACS: effective data access control for multiauthority cloud storage systems. IEEE Trans. Inf. Forensics Secur. 8 (11), 1790–1801. https://doi.org/10.1109/TIFS.2013.2279531.
Yang, K., Jia, X., Ren, K., 2015. Secure and verifiable policy update outsourcing for big data access control in the cloud. IEEE Trans. Parallel Distr. Syst. 26 (12), 3461–3470. https://doi.org/10.1109/TPDS.2014.2380373.
Yang, K., Han, Q., Li, H., Zheng, K., Su, Z., Shen, X., 2017. An efficient and fine-grained big data access control scheme with privacy-preserving policy. IEEE Internet Things J. 4 (2), 563–571. http://doi.org/10.1109/JIOT.2016.2571718.
Yu, S., Wang, C., Ren, K., Lou, W., 2010. Attribute based data sharing with attribute revocation. In: Proceedings of the 5th ACM Symposium on Information, Computer and Communication Security. ACM, pp. 261–270. https://doi.org/10.1145/1755688.1755720.

P.Praveen Kumar received the MTech degree in Computer and Information Technology from Manonmaniam Sundaranar University. He is currently doing PhD at National Institute of Technology, Tiruchirappalli. His research interests include Cloud Computing, Big Data, and Cryptography. He is a life member of the ISTE.

P .Syam Kumar received the MTech degree in Computer Science and Technology from Andhra University and PhD degree in Computer Science from Pondicherry University. He is an Assistant professor in Institute for Development and Research in Banking Technology (IDRBT), Hyderabad. His research interests are in the area of Cloud Computing, Security and Privacy, Cryptography and IoT. He is a member of the IEEE.

P .J.A.Alphonse received the MTech degree in computer science from Indian Institute of Technology, Delhi and the PhD degree in Mathematics & Computer Science from National Institute of Technology, Tiruchirappalli. He is currently working as Associate professor in National Institute of Technology, Tiruchirappalli. His research interests include Graph Theory and its Algorithms, Wireless and Ad hoc Networks, Cryptography and Network Security. He is a life member of the ISTE and ISC.